

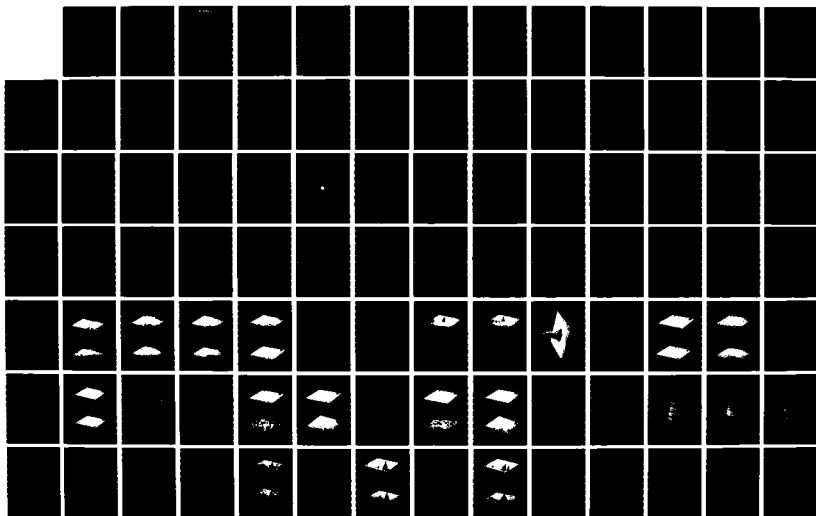
AD-A170 652

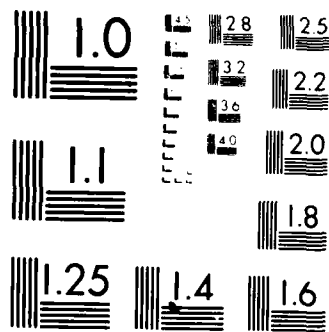
A NEW METHOD OF SYNTHETIC APERTURE RADAR IMAGE  
RECONSTRUCTION USING MODIF. (U) ILLINOIS UNIV AT URBANA  
COORDINATED SCIENCE LAB M D DESAI AUG 86  
UNCLASSIFIED UILU-ENG-86-2225 N00014-84-C-0149

1/2

F/G 17/9

NL





MICROCOPY RESOLUTION TEST CHART  
NATIONAL BUREAU OF STANDARDS-1963-A

12

# COORDINATED SCIENCE LABORATORY

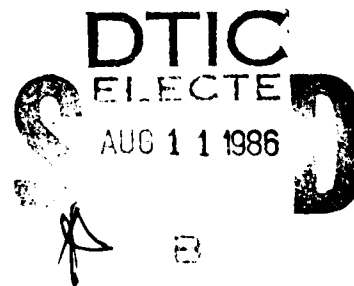
*College of Engineering*

**AD-A170 652**

## **A NEW METHOD OF SYNTHETIC APERTURE RADAR IMAGE RECONSTRUCTION USING MODIFIED CONVOLUTION BACK-PROJECTION ALGORITHM**

**Mita Dinkarraai Desai**

DTIC FILE COPY



**UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN**

Approved for Public Release. Distribution Unlimited.

SECURITY CLASSIFICATION OF THIS PAGE

## REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified			1b. RESTRICTIVE MARKINGS None	
2a. SECURITY CLASSIFICATION AUTHORITY N/A			3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release, distribution unlimited.	
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A				
4. PERFORMING ORGANIZATION REPORT NUMBER(S) UILU-ENG-86-2225			5. MONITORING ORGANIZATION REPORT NUMBER(S) N/A	
6a. NAME OF PERFORMING ORGANIZATION Coordinated Science Laboratory, Univ. of Illinois		6b. OFFICE SYMBOL (If applicable) N/A	7a. NAME OF MONITORING ORGANIZATION Office of Naval Research	
6c. ADDRESS (City, State and ZIP Code) 1101 W. Springfield Avenue Urbana, Illinois 61801			7b. ADDRESS (City, State and ZIP Code) 800 N. Quincy Arlington, VA 22217	
8a. NAME OF FUNDING/SPONSORING ORGANIZATION JSEP		8b. OFFICE SYMBOL (If applicable) N/A	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER Contract #N00014-84-C-0149	
8c. ADDRESS (City, State and ZIP Code) ONR 800 N. Quincy Arlington, VA 22217			10. SOURCE OF FUNDING NOS.	
11. TITLE (Include Security Classification) A New Method of Synthetic Aperture Radar Images Reconstruction Using Modified Convolution Back-Projection Algorithm			PROGRAM ELEMENT NO. N/A	TASK NO. N/A
12. PERSONAL AUTHOR(S) Desai, Mita D.			WORK UNIT NO. N/A	
13a. TYPE OF REPORT Technical		13b. TIME COVERED FROM _____ TO _____	14. DATE OF REPORT (Yr., Mo., Day) August 1986	
15. PAGE COUNT 92				
16. SUPPLEMENTARY NOTATION N/A				
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	SUB. GR.	Synthetic aperture radar, radar imaging, convolution back-projection reconstruction.	
19. ABSTRACT (Continue on reverse if necessary and identify by block number)				
<p>Synthetic Aperture Radar (SAR) image reconstruction falls into the class of inverse (deconvolution) problems. A spotlight mode SAR system obtains projections of the ground area at various look angles. The image of the ground area is then reconstructed from this set of projections.</p> <p>The Convolution Back-Projection (CBP) algorithm is widely used technique in Computer Aided Tomography (CAT). In this work the CBP algorithm was modified so that it can be applied to image reconstruction from SAR data. A quantitative evaluation using extensive computer simulation of the CBP algorithm for spotlight mode SAR is presented. Its performance is then compared with the FFT method with respect to parameters such as multiplicative noise ratio (MNR), central processing unit CPU time, and computational complexity. It has been shown that the CBP SAR algorithm is a high quality reconstruction method that can be implemented in real-time. The parallelism inherent in the algorithm can be exploited for parallel architectures that are suitable for VLSI implementation. Samples from each projection can be summed into the proper</p> <p>(Continued on reverse)</p>				
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED UNLIMITED <input checked="" type="checkbox"/> SAME AS RPT <input type="checkbox"/> DTIC USERS <input type="checkbox"/>			21. ABSTRACT SECURITY CLASSIFICATION Unclassified	
22a. NAME OF RESPONSIBLE INDIVIDUAL			22b. TELEPHONE NUMBER (Include Area Code)	22c. OFFICE SYMBOL None

19. ABSTRACT (cont.)

pixels in the final image, as required by the back-projection operation, so that the final image is produced shortly after the final projection has arrived.

A NEW METHOD OF SYNTHETIC APERTURE RADAR IMAGE RECONSTRUCTION  
USING MODIFIED CONVOLUTION BACK-PROJECTION ALGORITHM

BY

MITA DINKARRAI DESAI

B.S. Tennessee State University, 1979  
M.S., University of Illinois, 1981

THESIS

Submitted in partial fulfillment of the requirements  
for the degree of Doctor of Philosophy in Electrical Engineering  
in the Graduate College of the  
University of Illinois at Urbana-Champaign, 1985



DTIC  
ELECTE  
AUG 11 1986  
S D  
B

Urbana, Illinois

Accepted for	✓
NTIS	
EDS	
U	
D	
Dis	
A-1	

## A NEW METHOD OF SYNTHETIC APERTURE RADAR IMAGE RECONSTRUCTION USING MODIFIED CONVOLUTION BACK-PROJECTION ALGORITHM

Mita Dinkarra Desai, Ph.D.  
Department of Electrical and Computer Engineering  
University of Illinois at Urbana-Champaign, 1985

Synthetic Aperture Radar (SAR) image reconstruction falls into the class of inverse (deconvolution) problems. A spotlight mode SAR system obtains projections of the ground area at various look angles. The image of the ground area is then reconstructed from this set of projections.

The Convolution Back-Projection (CBP) algorithm is a widely used technique in Computer Aided Tomography (CAT). In this work CBP algorithm has been modified so that it can be applied to image reconstruction from SAR data. A quantitative evaluation using extensive computer simulation of the CBP algorithm for spotlight mode SAR is presented. Its performance is then compared with the FFT method with respect to parameters such as multiplicative noise ratio (MNR), central processing unit CPU time, and computational complexity. It has been shown that the CBP SAR algorithm is a high quality reconstruction method that can be implemented in real-time. The parallelism inherent in the algorithm can be exploited for parallel architectures that are suitable for VLSI implementation. Samples from each projection can be summed into the proper pixels in the final image, as required by the back-projection operation, so that the final image is produced shortly after the final projection has arrived.

## ACKNOWLEDGEMENTS

I wish to express my sincere appreciation and gratitude to Professor W. K. Jenkins, my dissertation advisor, for his invaluable guidance, continuing encouragement and patience during the course of my graduate studies. His friendly and optimistic attitude has been great source of inspiration over the past few years. I would like to thank Professor D. C. Munson Jr., for his patience and useful comments. I would also like to thank Professors T. N. Trick and J. Patel for being members of my dissertation committee, and Professors Van Valkenburg, J. B. Cruz, J. D. Dyson and W. R. Perkins for their encouragement and understanding.

I would like to thank Santosh Abraham, Praseon Tiwari, Joseph Rahmeh and other members of the Coordinated Science Laboratory for many helpful suggestions; Rosemary Wegeng and my office mates Eric Diethorn and Bruce Mather for their friendship. It is a pleasure to express my deepest appreciation to my good friend Professor Vasant Rao for his continued support and understanding. I wish to thank Subhasis Laha and Atul Jain for their friendship and their faith in me. I will always treasure the friendship of Susheel Chandra who made me see the lighter side of life. This research was funded in part by the Joint Services Electronics Program.

Finally, I wish to thank my parents for their everlasting love, encouragement, patience, and support. They have always been a great source of inspiration to me. This thesis is dedicated to both of them.

## TABLE OF CONTENTS

CHAPTER	Page
1. INTRODUCTION .....	1
2. OVERVIEW OF THE SIGNAL PROCESSING CONCEPTS FOR SAR .....	7
2.1 The SAR System .....	7
2.2 The Tomographic Interpretation of SAR .....	13
2.3 The Conventional Method .....	16
3. CONVOLUTION BACK-PROJECTION ALGORITHM AS APPLIED TO SAR .....	19
3.1 The Convolution Back-Projection Algorithm for SAR .....	19
3.2 The Implementation for SAR .....	21
3.2.1 CBP Algorithm as Projectional Reconstruction .....	23
3.2.2 CBP Algorithm as Point-by-Point Reconstruction .....	23
3.3 Signal Processing Aspects of the Algorithm .....	26
3.4 Complexity of the Algorithm .....	27
4. ARCHITECTURE FOR SAR IMAGE RECONSTRUCTION USING CBP .....	28
4.1 Nature of Computation .....	28
4.2 Architectural Consideration .....	30
4.2.1 One Processor Per Pixel .....	30
4.2.2 One Processor Per Column .....	33
4.3 Feasibility of Architecture .....	40
5. SIMULATION AND RESULTS .....	44
5.1 Computer Simulation for Ideal Data .....	45
5.1.1 Illustration of Algorithmic Characteristics .....	48
5.1.2 Signal Processing Aspects .....	58
5.1.3 Comparisons with FFT Based Method .....	62
5.1.4 Point Target Response .....	72
5.2 Non-Ideal Data .....	78
6. CONCLUSIONS .....	85
REFERENCES .....	88
VITA .....	93

## CHAPTER 1

### INTRODUCTION

There are many existing imaging systems with their own characteristic advantages and limitations. Synthetic Aperture Radar (SAR) is an active microwave imaging system with a side-looking antenna, transmitting pulses and receiving the returned microwave echoes [6.7.19]. The microwave sensors used for SAR are unique in their ability to penetrate cloud cover and rain. This makes the use of microwave system more advantageous than its optical and infrared counterparts. These radars not only work during bad weather but also observe certain phenomena not observed by optical imaging systems. For example, they can provide information about temperature, moisture, texture, and electrical properties of the terrain [50]. For these reasons, SAR is used as a very effective microwave imaging technique for ground mapping, remote sensing and surveillance and for producing high resolution terrain images.

SAR imaging systems yield two-dimensional images which are usually called the range and the azimuth. **Azimuth** is the direction of the flight path of the plane. The direction orthogonal to azimuth called the **range** is the direction of transmitting and receiving the signal. Throughout the flight interval, the antenna is steered to illuminate a fixed area of interest enabling one to synthesize a long linear array of antennas. Such an operating mode is known as the **spot-light mode**. As a result, the along track resolution is made much finer and is not limited by the actual antenna beam width [34.39].

High resolution in range (cross track) direction is achieved by means of pulse compression waveforms. One commonly used waveform is linear FM, often called "chirp" [34.13]. The range resolution is obtained from accurately measured time-delay information of the echoes. During the entire flight interval, the signal is transmitted at discrete points, designated by the pulse repetition frequency. The received data are then appropriately

recorded and processed.

The returned echoes are mixed with a reference signal, lowpass filtered, sampled and recorded digitally. Munson, Jenkins and O'Brien [45] have shown that the spot-light mode SAR can be viewed as a parallel beam tomographic process with the image reconstruction based on the projection slice theorem. Consequently, it has been shown that the pre-processed SAR returns are really polar samples of the two-dimensional Fourier transform of the illuminated region [5,54]. Suppose the complex ground reflectivity function is  $g(x,y)$ , and its 2-D continuous-space Fourier transform is  $G(\omega,\theta)$ . The SAR system then produces samples of  $G(\omega,\theta)$ , namely,  $G(n\Delta\omega,m\Delta\theta)$ ,  $n,m = 1,2,\dots,N$  where  $N$  is the number of samples in each direction. Here the range of  $\omega$  is proportional to the signal bandwidth, and the range of  $\theta$  is proportional to the length and position of flight. Therefore, SAR is a process that produces discrete frequency Fourier data from the continuous space ground reflectivity function. From these polar samples in a small region of the Fourier domain, the image of the ground reflectivity,  $\tilde{g}(x,y)$  has to be reconstructed. Hence, the reconstruction can be viewed as a process that obtains a discrete space reflectivity function from the discrete frequency Fourier data. In summary, the SAR system falls in the third category while the reconstruction falls in the fourth among the following four categories<sup>1</sup> of problems relating space and frequency as considered in [27].

1. continuous space - continuous frequency
2. discrete space - continuous frequency
3. continuous space - discrete frequency
4. discrete space - discrete frequency

Given the samples of the two-dimensional Fourier data, i.e., the spatial frequency data, over a finite region, the image of the ground reflectivity,  $\tilde{g}(x,y)$ , needs to be

---

<sup>1</sup>Private communication with Dr. L. Huang and Dr. W. K. Jenkins

computed. This involves inverting the Fourier data to obtain the spatial image. Therefore, SAR, like Computer Aided Tomography (CAT), sonar, X-ray crystallography and many other problems, falls into the category of inverse problems.

SAR image reconstruction is an inverse problem with incomplete observations. Since the plane flies along a straight line, even a few degrees could translate into a long flight distance. Therefore, the measurements are possible only for a rather small angular range (usually less than 10 degrees). In this way the SAR system is similar to limited angle tomography. Reconstructing the two-dimensional SAR image involves inverting the discrete set of angularly limited set of measured data. Despite many similarities between the two processes, the major difference is the fact that SAR is a coherent process. Both magnitude and phase are measured in SAR as opposed to only intensity measurements in tomography.

The first SAR system operating at X band was developed at the University of Illinois in 1953. In the late fifties, with battlefield surveillance in mind, a major project was started at University of Michigan [3]. In 1957, the Willow Run Laboratories of the University of Michigan produced the first focused SAR image using an optical processor. Since then SAR has been applied to many civilian areas, such as in radar astronomy, image planetary surfaces [14], oceanography (to gain information on surface waves, currents, and oil slicks [2]), and in geological mapping [42]. The usefulness of SAR in various fields has made it a major technological tool.

SAR image reconstruction and the associated processing algorithms have evolved considerably over last three decades. Starting from the analog optical methods, more sophisticated digital processing algorithms have been developed. A widely used method inverse Fourier transforms the given data and is similar to optical methods. This method requires interpolating the data from the polar grid onto a uniformly spaced rectangular grid in order

to apply the inverse 2-D FFT. One of the disadvantages of this system is that it requires the use of an accurate and efficient 2-D polar to rectangular interpolator [47].

The major bottlenecks in SAR processing are the inaccurate and expensive interpolators (polar to rectangular) and the lack of a very fast, real-time algorithm. The present research was stimulated by the urgent need to develop a new, highly efficient, real-time algorithm along with an appropriate VLSI architecture to produce accurate SAR images at a lower cost and a much faster rate, on a large scale basis.

The preliminary aim of this work was to develop new methods of SAR image reconstruction. This has led to a Convolution Back-Projection (CBP) algorithm modified for SAR. The basic CBP algorithm has been in existence for quite a while and has been used mainly for tomographic image reconstruction [53.1.30]. The results in this dissertation show that this algorithm can be very easily used for SAR image reconstruction with appropriate modifications. This eliminates the overhead of polar-to-rectangular interpolation and the associated approximation. Also, the inherent parallelism in the algorithm can be exploited using parallel VLSI architectures.

With the intention of defining some aspects of SAR along with the associated geometry, Chapter 2 starts with SAR basics along with the description of the conventional method of SAR image reconstruction. Since this algorithm is based on the tomographic interpretation of SAR, a brief review of this concept is also presented. The notation and formulas defined here will be used throughout the entire thesis.

Chapter 3 begins with the description of the Convolution Back-Projection algorithm (CBP) and the details of the modifications necessary for the SAR data. The difference between applying the CBP method to reconstructing SAR images as opposed to tomographic images is explained. Furthermore, several useful characteristics inherent in the CBP technique are discovered. Some of these characteristics include the projectional interpretation,

the point-by-point image reconstruction, and the inherent parallelism in the algorithmic structure. The projectional interpretation results in the capability of producing images on-line as the projectional measurements are made available. This feature of *image-evolution* may be advantageous in military situations since one need not wait till the end of the flight path before discovering, at least approximately, the location of important targets. The point-by-point image reconstruction enables the user to reconstruct any portion of the entire image as desired. If the approximate target location is known *a priori*, then this allows the user to zero-in on the region of the target only, thus resulting in substantial savings in computation time. This chapter also examines some of the signal processing aspects such as windowing and interpolation used to improve the quality of the images produced. Finally, the computational complexity of the CBP algorithm is discussed and shown to be of  $O(N^3)$  in the worst case, where  $N \times N$  is the size of the image.

Application of the CBP algorithm to SAR brings out its latent characteristics like the parallelism and real-time processing capability. Chapter 4 discusses these implementation issues of the CBP algorithm and its architectural implications. An architecture exploiting the inherent parallelism and pipelining features of the algorithm is suggested and its feasibility for VLSI technology is established.

The modified CBP algorithm is implemented in the form of a computer program. The characteristics of the algorithm are illustrated by several examples and the results are presented in Chapter 5. The performance of the CBP algorithm is compared with its conventional counterpart, the 2-D FFT based method. The comparison is made by considering the qualitative measure as the multiplicative noise ratio (MNR) and the total computation time. The results presented in Chapter 5 indicate that that CBP is in fact an accurate and highly desirable algorithm for SAR image reconstruction.

Chapter 6 concludes with discussions on the further needs of study for perfecting this algorithm and its implementation.

## CHAPTER 2

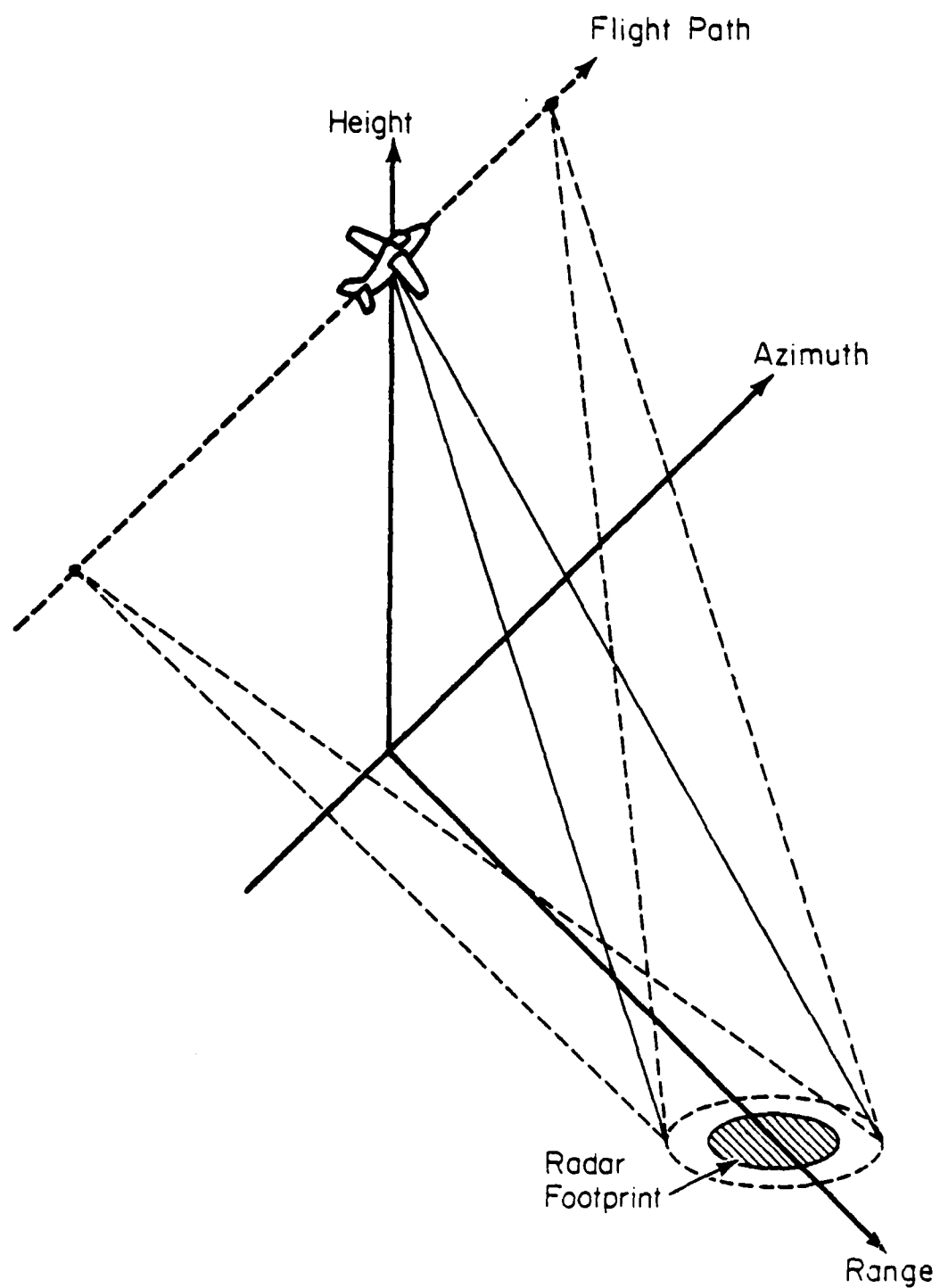
### OVERVIEW OF THE SIGNAL PROCESSING CONCEPTS FOR SAR

Synthetic Aperture Radar is a complex process involving many important concepts. Much of SAR literature is either classified or company proprietary. As a result unclassified published literature is incomplete, and in some cases considerably out of date. In this chapter, the basic SAR system and its processing technique is reviewed. A review of the conventional method of SAR data processing, which will be referred to as the Direct Fourier Method (DF) and important concepts like the tomographic interpretation of SAR, are also presented since these concepts will often be referred to in the chapters to follow. Review, along with some insight into the SAR problem, will be helpful to the readers not familiar with SAR.

#### 2.1 The SAR System :

Terrain images are produced by properly processing returned microwave echoes received via a side-looking conventional antenna carried on an airplane or satellite [34]. The geometry for the data collection in spotlight mode SAR is shown in Figure 2.1. The region of the ground illuminated by the radar beam is known as the radar footprint. The shaded area of the footprint corresponds to the region covered by the 3dB beam-width of the far field antenna pattern. Let the reflectivity of the ground patch be a complex quantity  $g(x,y)$ . Returned data will contain the sum of the complex reflectivity of the ground along the parallel lines. The final aim of a SAR processing system is to compute the magnitude of the reflectivity from the returned echoes.

The azimuth resolution for a conventional radar is limited by the antenna beam width. If the physical aperture of the antenna is  $D$ , and if the signal wavelength is  $\lambda$ , then the azimuth resolution will be of the order  $\delta_{az} = R\lambda/D$  at the range  $R$  [31]. To achieve a fine resolution would require the antenna aperture to be too large to be physically carried on an aircraft or



FP-6795

Figure 2.1 : SAR geometry.

spacecraft. SAR circumvents this problem by linearly synthesizing the antenna, i.e., assuming the position of an element in a linear array as the plane flies along its trajectory. As a result, the along-track resolution is no longer limited by the actual antenna beam width [21]. If the antenna is steered so that a fixed area of interest is illuminated throughout the entire collection interval, then the resolution is made even finer. This specialized form of operation is called the **spotlight mode SAR**.

High resolution in range (cross-track resolution) is achieved by pulse compression using waveforms like the linear FM waveform (chirp) [21]. A distinguishing feature of a pulse compression waveform is that its time bandwidth product is much larger than unity. It provides large bandwidth without large peak power. The analytic form of the expression for the linear FM chirp is

$$S(t) = \begin{cases} e^{j(\omega_0 t + \frac{\gamma}{2} t^2)} & -T/2 < t < T/2 \\ 0 & \text{otherwise} \end{cases} \quad (2.1.1)$$

where,  $\omega_0$  is the carrier frequency and  $\gamma/2$  is the FM rate.

The returned signal from a point target  $(x_t, y_t)$  as in Figure 2.2, will be delayed in time by the round-trip delay from the radar to a target. When this returned signal is mixed with a signal simulating the return from the chosen reference point, the phase of the resulting signal will have a term proportional to the time delay. The time delay of the signal is converted into a frequency component linearly related to the distance from the target to the reference point. The fact that the frequency depends linearly on the delay difference, allows the location of the target in time, or range direction with respect to the chosen reference point. Similarly, the phase function of the return signal will also contain Doppler shifts. This Doppler term varies from pulse to pulse but can be ignored as far as range is concerned. Each pulse gives a sample of this difference Doppler shift. When the return is mixed with a reference, the resultant dif-

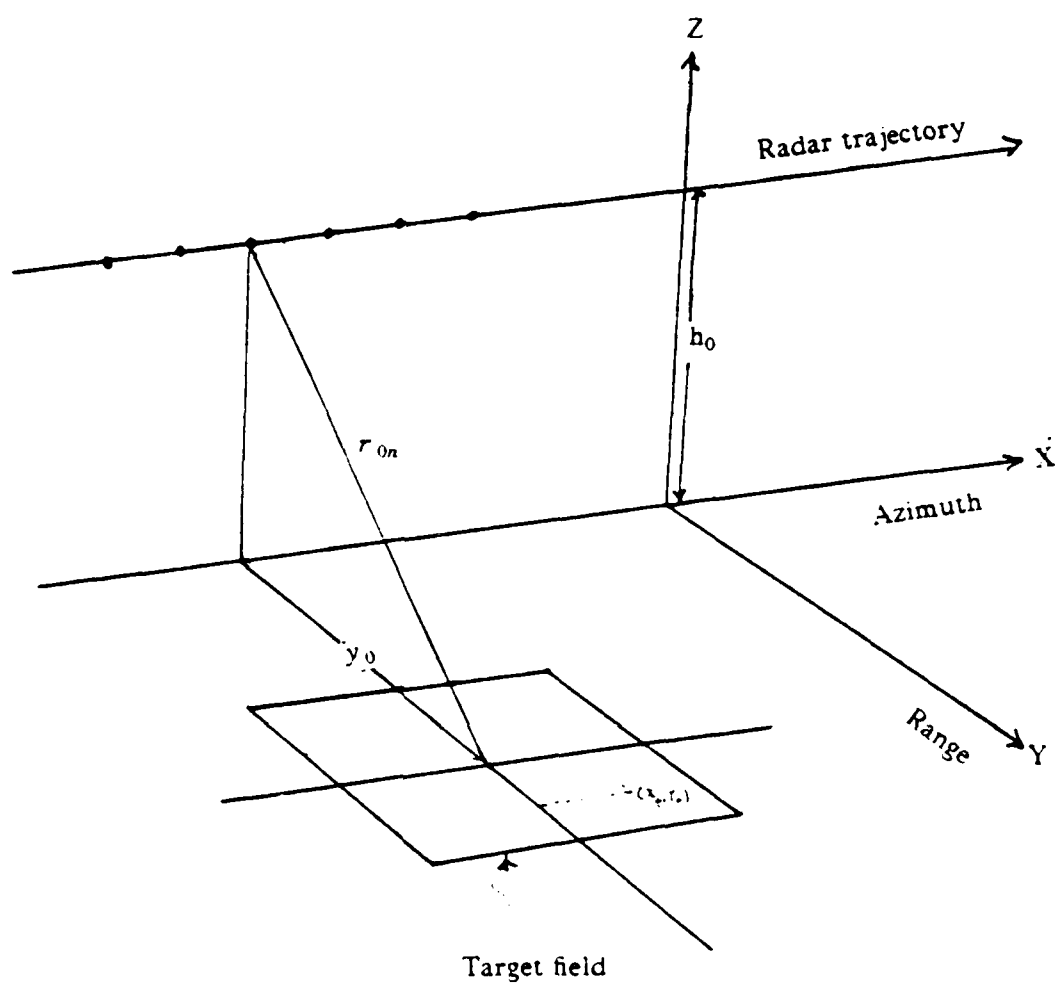


Figure 2.2 : SAR data collection geometry.

ferential Doppler frequency will give the differences in the azimuth location.

The following reasonable assumptions will be used later in the chapter to make the analysis of the system easier. These assumptions are typical of any SAR system :

- 1) The target is in the far field of the radar.
- 2) The target can be accurately modelled by independent point scatterers.
- 3) The target region is uniformly illuminated
- 4) The phase centers of the targets remain constant over the entire angular range, i.e., the returns are independent of the angle of illumination, and
- 5) The reflectivity,  $g(x,y)$  remains constant over the bandwidth of the signal.

Each return is mixed with the reference signal and lowpass filtered. Following these processes, samples of each return are recorded along a radial line. Successive returns are recorded on the radial line with varying polar angle  $\theta$ , as illustrated in Figure 2.3. The angular increment corresponds to the angle between the signal receiving point along the flight path and the reference point on the ground. In other words the angular increment is a function of the pulse repetition frequency of the radar.

The main idea behind the data gathering process can be described as a radar transmitting a chirp, mixing, low-pass filtering, and recording each received return for further processing. The difficulties with processing this data set are

- 1). The data are on a polar grid.
- 2). The data are only known within a small region of the transform domain.

With the conventional methods of processing, (1) requires polar to rectangular interpolation; item (2) effects the limit on the achievable resolution and therefore the quality of the final image.

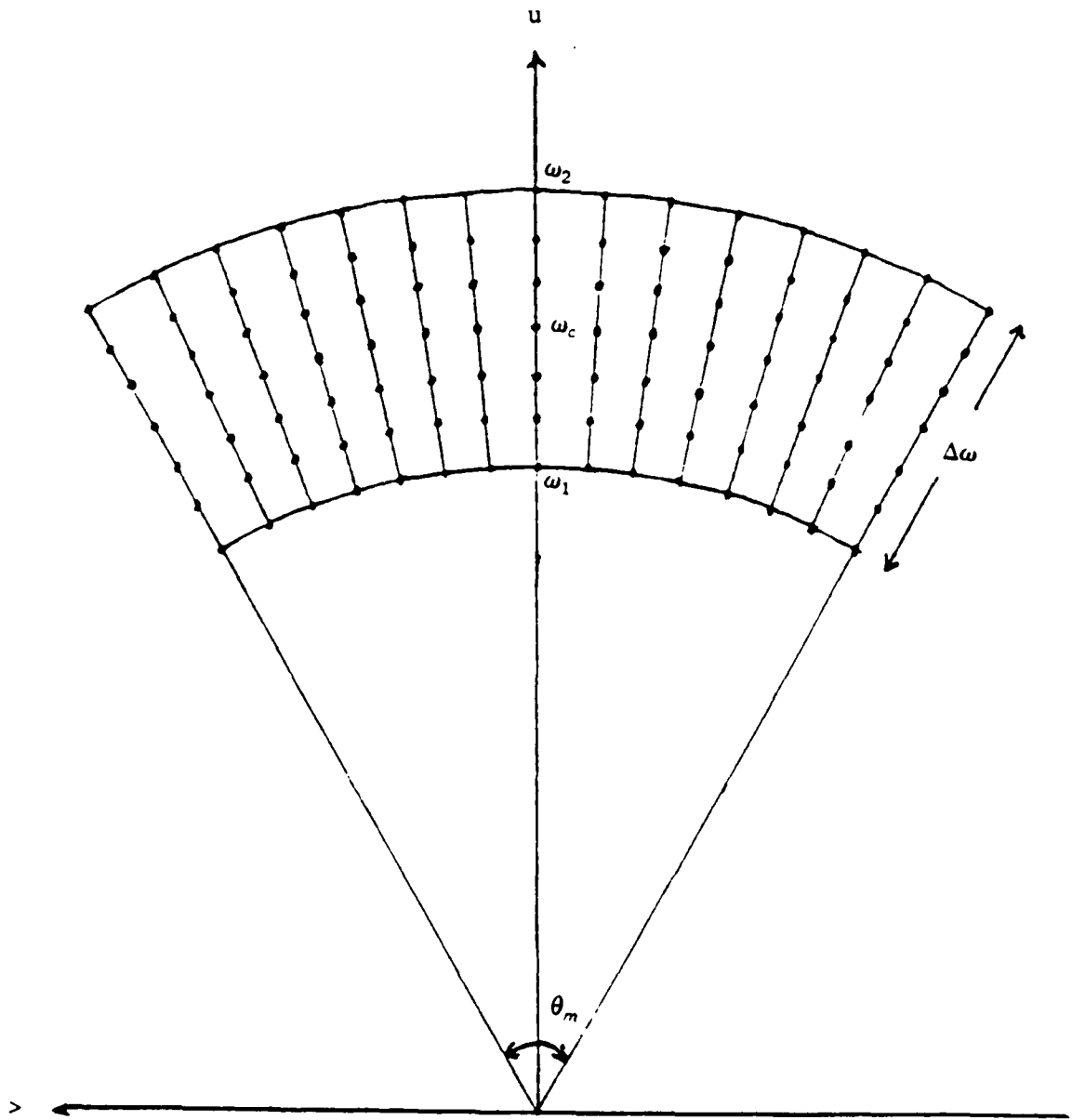


Figure 2.3 : Data recording raster for SAR

For such a data set of a ground field, the achievable resolution with SAR is [55]

$$\delta_{az} = \lambda_0 r_{00}/2L \quad (2.1.2)$$

and

$$\delta_{range} = c \pi r_{00}/\gamma y_0 T_e, \quad (2.1.3)$$

where  $T_e$  is the effective pulse duration in seconds,  $\gamma$  is the FM sweep rate in rad/sec, and  $L$  is the synthetic aperture length. In equation (2.1.3)  $y_0$  is the distance from the radar ground track to the reference point as shown in Figure 2.2 in meters. However, factors such as the coherency of the microwave source, stability of the flight path and atmospheric turbulence will cause phase errors resulting in defocussing and degradation of the resolution.

## 2.2 The Tomographic Interpretation of SAR

The well-known interpretation of SAR is that it is a one-dimensional holographic process [29,37,40]. The most recent interpretation is that of SAR as a parallel beam tomographic process. SAR system measures the returns of the target field from different angular points along its trajectory. The notion of different angular measurements leads one to question the similarity of the signal processing aspects of the tomographic and the SAR processes. This different perspective on SAR gives an excellent insight into the SAR process [48]. It is the basis of the proposed CBP algorithm. The details of this concept first published in a paper by Munson, Jenkins and O'Brien [47] are reviewed below.

If the linear FM pulse described in Equation 2.1.1 is transmitted, then the returned signal from a point target located at  $(x_0, y_0)$  is,

$$r_0(t) = g(x, y) S(t - 2R_0/c). \quad (2.2.1)$$

Where  $R_0$  is the distance from the plane to the point  $(x_0, y_0)$ . The equidistant points from the plane to the target field will lie on a curve as shown in Figure 2.4 by the dotted lines. These curves can be approximated by a straight line, when  $R \gg L$ , using the assumption 1 of the

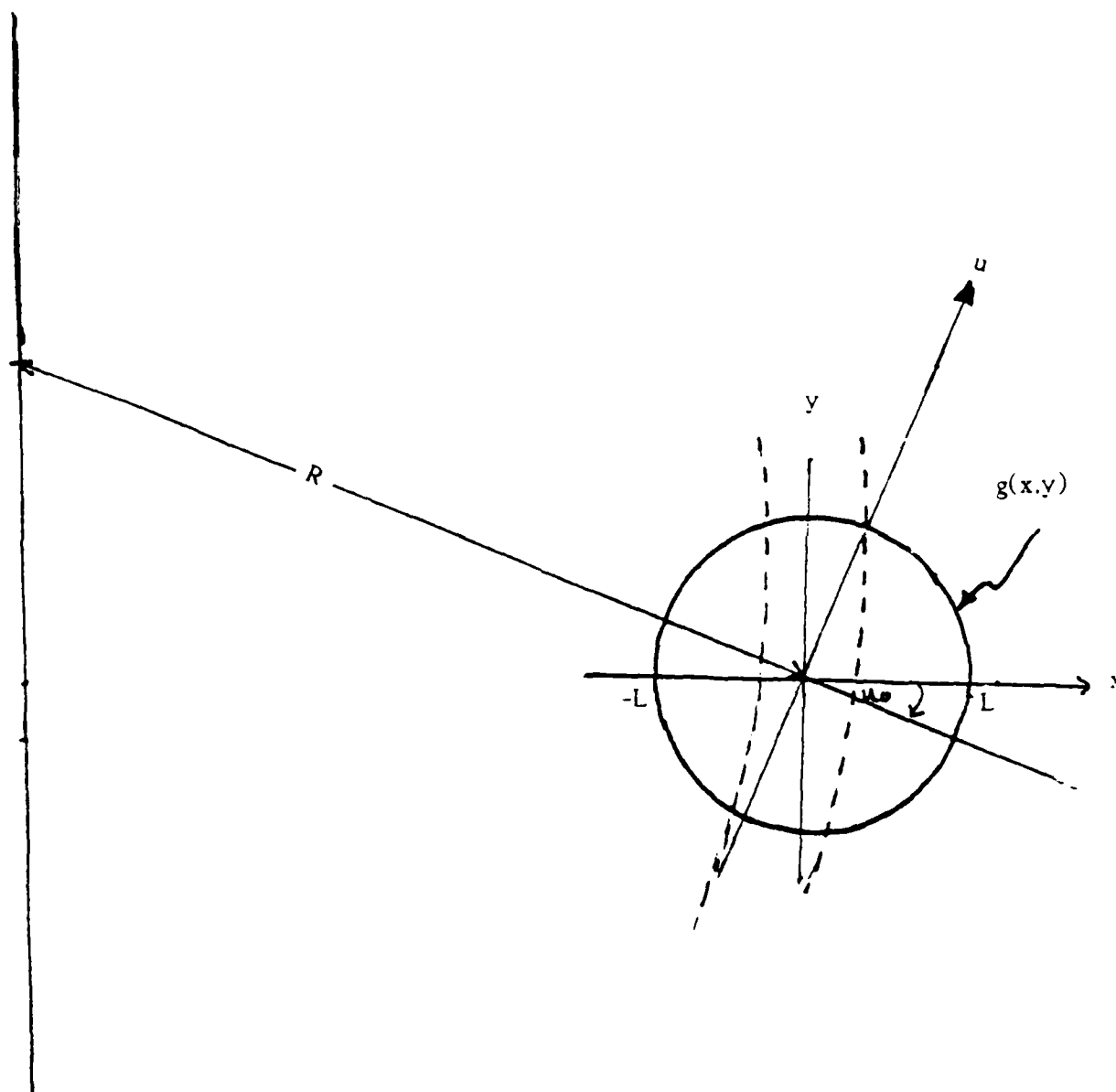


Figure 2.4: Equidistant points lie on an arc and since  $R \gg L$ , these can be approximated by a straight line.

previous section.

Returns for a level curve will be the sum of the reflectivity of all the points scattered on this line. Therefore, the return from a line  $u=u_0$  is given by:

$$r_u(t)|_{u=u_0} = P_u(u_0) S(t - 2(R_0 + u_0)/c) \quad (2.2.2)$$

where,  $P_u(u_0)$  is the projection of the ground patch evaluated at  $u=u_0$ . The projection is defined to be

$$P_u(u) = \int_{-\infty}^{\infty} g(ucos\theta - v sin\theta, u sin\theta + v cos\theta) dv. \quad (2.2.3)$$

Therefore, returns from the entire ground patch can be written as

$$r_0(t) = \int_{-L}^{+L} P_u(u) S(t - 2(R_0 + u)/c) du \quad (2.2.4)$$

$$\text{for } -T/2 + \tau_0 + \frac{2L}{c} \leq t \leq T/2 + \tau_0 - \frac{2L}{c}$$

Here  $\tau_0 = 2R_0/c$  is the two way time delay from the center of the ground patch. Mixing the returned signal,  $r_0(t)$ , with the reference chirp

$$S_{ref} = e^{j(\omega_0(t-\tau_0) + \gamma(t-\tau_0)^2)} \quad (2.2.5)$$

and low-pass filtering results in

$$C_u(t) = \int_{-L}^{+L} P_u(u) \exp(j \frac{4\gamma u^2}{c^2}) \exp[-j \frac{2}{c}(\omega_0 + 2\gamma(t-\tau_0)u)] du. \quad (2.2.6)$$

The effect of the quadratic phase can be neglected since the length of the data collection is small compared to the distance  $R_0$ , thus yielding.

$$\bar{C}_u(t) = \int_{-L}^{+L} P_u(u) \exp[-j \frac{2}{c}(\omega_0 + 2\gamma(t-\tau_0)u)] du. \quad (2.2.7)$$

The above Equation can be interpreted as the Fourier Transform of the projection of the ground patch  $p_u(u)$ .

$$\bar{C}_u(t) = P \left[ \frac{2}{c}(\omega_0 + 2\gamma(t-\tau_0)) \right] \quad (2.2.8)$$

$$\text{for } -T/2 + \tau_0 + \frac{2L}{c} \leq t \leq T/2 + \tau_0 - \frac{2L}{c}$$

In other words,  $\tilde{C}_\theta(t)$  is a slice at the angle  $\theta$  of the 2-D Fourier transform of  $g(x, y)$ . Rewriting Equation (2.2.8),

$$\tilde{C}_\theta(t) = P_\theta(\omega) \quad \text{for } \omega_1 \leq \omega \leq \omega_2$$

where

$$\omega_1 = \frac{2}{c} \left[ \omega_0 - \gamma T + \frac{4\gamma L}{c} \right], \quad \text{and} \quad \omega_2 = \frac{2}{c} \left[ \omega_0 + \gamma T - \frac{4\gamma L}{c} \right].$$

N such equidistant projections give the data region shown in Figure 2.3. Note that the projections are known only from  $\omega_1$  to  $\omega_2$ .

### 2.3 The Conventional Method

The more conventional Direct Fourier method, involves reconstructing the image by means of the 2-D inverse FFT. First the algorithm assumes that the information outside the annular region is zero. The data in reality are not zero; they are unknown. Within this annular region, a square is inscribed. The resolution of the image is a function of the size of the square.

Given the Fourier samples on a polar raster with coordinates  $(\omega, \theta)$ , it is first interpolated onto a cartesian raster with coordinates  $(u, v)$  as shown in Figure 2.5 and then a 2-D inverse FFT is applied. The interpolation step is a critical part of the algorithm. The conventional method is the Standard Polar Format (SPF) algorithm [47]. Basically it is a separable windowed sinc interpolation kernel. The sinc kernel is truncated by a window such as a Hamming window. First it is applied in the radial (range) direction, producing a key-stone grid, and then applied in the azimuth direction to produce the final rectangular grid. Note that interpolation from nonuniform sampling is required and that the SPF algorithm can perform this task accurately and conveniently. Conversion from polar to rectangular coordinates also results in the changing the sampling rate. The square is shifted to the origin of the  $(u, v)$  plane and the data are then processed by an array processor to inverse transform the data using a 2-D FFT. The

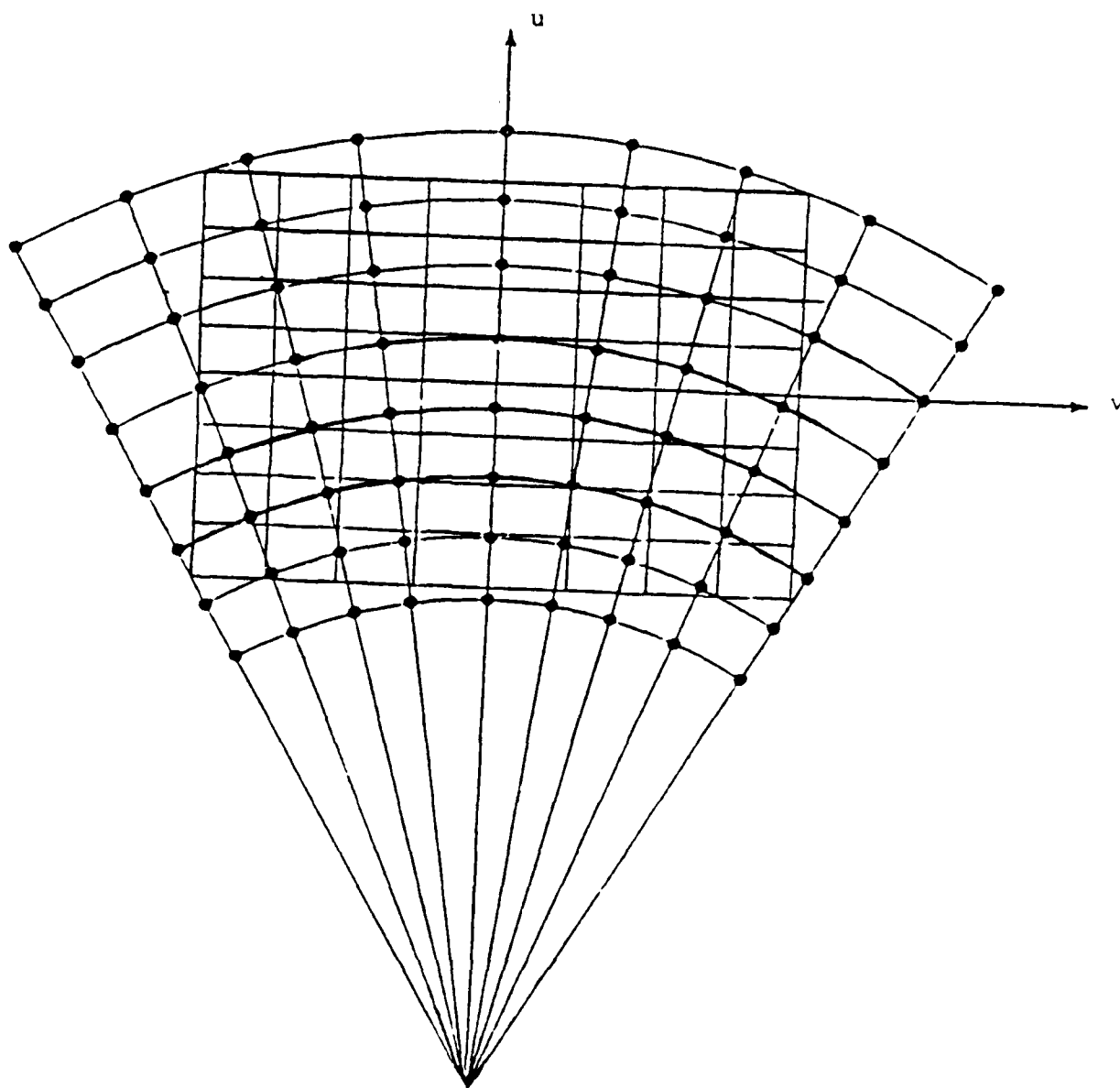


Figure 2.5 : Polar to rectangular conversion.

final image is then displayed as the magnitude of the resulting 2-D array.

This is a very straightforward algorithm. The complexity of this algorithm is  $P(N) + O(N^2 \log_2 N)$ ; where  $P(N)$  is the computational complexity of the polar to rectangular interpolator and  $O(N^2 \log_2 N)$  is the computational complexity of the 2-D FFT. If the  $O(P(N))$  is less than  $O(N^2 \log_2 N)$  then the computational complexity of the entire algorithm is of  $O(N^2 \log_2(N))$ . Therefore, the speed and the availability of the fast 2-D FFT processors make this algorithm a very attractive one. However, a major difficulty with this algorithm is that it requires the use of an accurate 2-D polar to rectangular interpolator. This polar-to-cartesian interpolation must be accurate to prevent aliasing as well as spurious targets within the patch. Two-dimensional windows are applied to the Fourier domain data prior to transformation to reduce sidelobes and control the shape of the ideal point target response of the system.

## CHAPTER 3

### THE CONVOLUTION BACK-PROJECTION ALGORITHM AS APPLIED TO SAR

It has been shown that image reconstruction in Computer Aided Tomography (CAT) and spot-light mode synthetic aperture radar (SAR) are based on similar mathematical principles [47,48]. Subsequently, it was proposed that convolution back-projection (CBP) reconstruction [20,30], a very successful algorithm for CAT processing, could be also useful for SAR processing. This interpretation leads to the application of the CBP algorithm to the SAR data for coherent image reconstruction. In tomography, the projections are measured in spatial domain, while in SAR, the projectional data are recorded as slices in the Fourier domain. Therefore, the CBP algorithm needs some modification before it can be applied to SAR. This chapter examines the use of CBP reconstruction in SAR and describes modifications of CBP needed for the spotlight mode geometry. In the process, some of the latent characteristics of the CBP algorithm such as the projectional interpretation and the point-by-point reconstruction are observed.

#### 3.1 The Convolution Back-Projection Algorithm for SAR

The convolution back-projection (CBP) algorithm can be used in SAR for coherent image reconstruction [17,18]. The reconstruction procedure using CBP is derived here from the 2-D Fourier transform pairs,

Consider the complex ground reflectivity function  $g(x,y)$  with its 2-D Fourier transform  $G(u,v)$  defined as follows:

$$G(u,v) = \int \int_A g(x,y) e^{-j(ux+vy)} dx dy \quad (3.1)$$

$$g(x,y) = \int \int_L G(u,v) e^{j(ux+vy)} du dv \quad (3.2)$$

where  $A$  is the support of the function  $G(u,v)$  and  $L$  is the support of the function  $g(x,y)$ .

Let  $(\omega, \theta)$  represent the polar coordinates in the  $(u, v)$  plane. Therefore  $G(\omega, \theta)$  denotes the value of  $G(u, v)$  along a line at an angle  $\theta$  with the  $u$ -axis as in Figure 2.3. Therefore expressing Equation 3.2 in polar coordinates with  $|\omega|$  being the Jacobian, the reconstructed image  $\tilde{g}(x, y)$  will be

$$\tilde{g}(x, y) = \int_{-\theta_m/2}^{\theta_m/2} \int_{\omega_1}^{\omega_2} G(\omega, \theta) e^{j\omega(x\cos\theta + y\sin\theta)} |\omega| d\omega d\theta \quad (3.3)$$

A projection of  $g(x, y)$  at an angle  $\theta$ , is

$$p_\theta(t) = \int g(t, s) ds \quad (3.4)$$

where  $(t, s)$  denotes the  $(x, y)$ -coordinate system rotated by angle  $\theta$ . The one dimensional Fourier Transform of the projection is

$$\begin{aligned} P_\theta(\omega) &= \int p_\theta(t) e^{-j\omega t} dt \\ &= G(\omega, \theta) \end{aligned} \quad (3.5)$$

This is the well-known projection slice theorem, i.e., the Fourier transform of a projection of an object at an angle  $\theta$  is a slice of the Fourier transform of the object at the same angle. Applying this to equation 3.3, the SAR image reconstruction involves solving the following integral:

$$\tilde{g}(r, \phi) = \int_{-\theta_m/2}^{\theta_m/2} \int_{\omega_1}^{\omega_2} P_\theta(\omega) |\omega| e^{j\omega r \cos(\phi - \theta)} d\omega d\theta \quad (3.6)$$

where  $(r, \phi)$  represents the polar coordinates in the  $(x, y)$  plane.

The above formula for reconstruction suggests that from each projection,  $P_\theta(\omega)$ , the inner integral can be evaluated for each angle  $\theta$  yielding  $Q(t)$ , where  $t = r \cos(\phi - \theta)$ . The image  $g(r, \phi)$  can then be obtained by integrating the function  $Q$  over the look angle  $\theta$ . For the limited data region of Figure 2.3, the realization of the reconstruction algorithm may be accomplished via the following equations:

$$Q(t) = \int_{\omega_1}^{\omega_2} P_\theta(\omega) |\omega| e^{-j\omega t} d\omega \quad (3.7)$$

$$g(r, \phi) = \int_{-\pi/2}^{\pi/2} Q(r \cos(\phi - \theta)) d\theta. \quad (3.8)$$

Note that the parameter  $\omega$  has the dimension of spatial frequency. Also,  $t$  is the projectional domain variable with:  $\omega \longleftrightarrow t$ .

It must also be noted that in CAT reconstruction the input data provided are a collection of projections  $\{P_\theta(t)\}$ . Hence, for each angle  $\theta$ ,  $Q(t)$ , as defined in Equation 3.8, is obtained by convolving  $P_\theta(t)$  with the impulse response of a filter referred to as the *convolution* step and  $Q(t)$  is often called a "filtered projection". In SAR reconstruction, however, the input data given are a collection of the 1-D Fourier transform of the projection namely,  $P_\theta(\omega)$ , for each angle  $\theta$ . In this case, therefore,  $Q(t)$  is computed by performing an inverse Fourier transform of the product  $P_\theta(\omega) |\omega|$ . It is for this reason that SAR reconstruction can be treated as a dual of CAT reconstruction. The second step of computation described is similar in both processes. The combination of the two steps is referred to as the CBP algorithm. One of the key differences in applying CBP for SAR reconstruction as opposed to CAT reconstruction has just been explained above. A few other modifications necessary for applying this image reconstruction algorithm to SAR are presented in the following section.

### 3.2 The Implementation for SAR

In SAR the radar signal is modulated by a carrier frequency  $\omega_c$  and, therefore, the recorded projections are shifted by this carrier frequency. Each projection, therefore, must be shifted back by  $\omega_c$ . In a sense, this is equivalent to demodulating each projection by the carrier frequency. This operation requires substituting  $\omega$  by  $\omega - \omega_c$  in Equation 3.7. Another difference between CAT and SAR is that the given data set is discrete in SAR, i.e., for each angle  $\theta_i$ , the  $N_s$  samples  $\{P_{r_k}(\omega_i)\}$  of  $P_{r_k}(\omega)$  are provided. Therefore, the convolution step of Equation 3.7 simply reduces to a multiplication followed by a 1-D FFT computation. Let

$\{Q_k(t_m) : m = 1, 2, \dots, N_s\}$  be a set of samples of the function  $Q(t)$  computed from the projections at angle  $\theta_k$ . Then the modified equations are

**Convolution:**

$$\{Q_k(t_m)\} = F_1^{-1} \left\{ P_{\theta_k}(\omega_l - \omega_c) |\omega_l - \omega_c| \right\} \quad (3.9)$$

**Back-projection:**

$$\hat{g}(r, \phi) = \sum_i^{N_p} \hat{Q}(r \cos(\phi - \theta_k)) e^{j\omega_c r \cos(\phi - \theta_k)} \Delta\theta \quad (3.10)$$

The quantity  $\hat{Q}(r \cos(\phi - \theta_k))$  denotes the value of the function  $Q(t)$  obtained at  $t = r \cos(\phi - \theta_k)$  from the set of samples  $\{Q_k(t_m)\}$  via interpolation.

Thus, SAR image reconstruction is a two step process. Here each radial line in Figure 2.3 is considered to be a projection centered at  $\omega_c$ . Each projection is multiplied by a rectangular pulse of width  $\Delta\omega$ . It is then weighted, shifted to the origin, and inverse transformed via the 1-D FFT. Since convolution in the spatial domain is a product of Fourier Transforms in the frequency domain, the first operation is known as the convolution step. Convolution is efficiently implemented in the Fourier domain by simply multiplying by  $P_\theta(\omega - \omega_c)$  and  $|\omega - \omega_c|$ . As seen in Equation 3.7, the first step is simply the multiplication of each projection by the Jacobian, followed by the 1-D inverse FFT. The second step involves one-dimensional interpolation of  $Q$  for each point  $(x, y)$  or  $(r, \phi)$  and then appropriate back-projection to the corresponding angle of projection. The second step of back-projection, as characterized by Equation 3.8, is approximated by a summation. Intuitively, implementation of this step involves back smearing of each projectional values onto the image. Such an operation is executed for each projection. Thus, the operation of 2-D inverse transforming is done directly on the polar-formatted data in two steps.

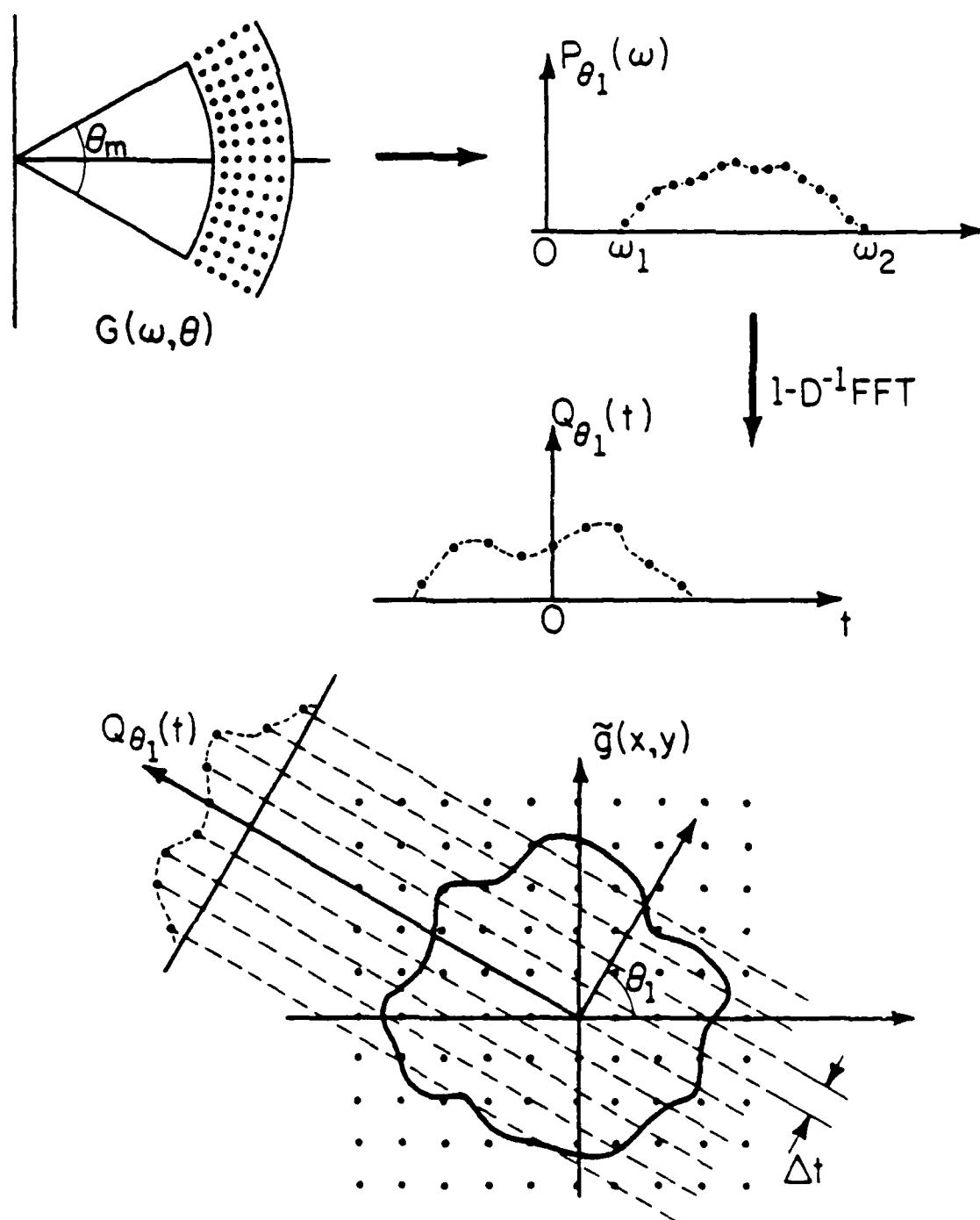
Given the samples  $\{G(\omega_l, \theta_k)\}$  of the 2-D Fourier transform of the ground reflectivity function over the annular region, the spatial function  $\tilde{g}(r, \phi)$  is computed. The actual computation is performed for discrete image pixels denoted by  $(x_i, y_j)$  or simply  $(i, j)$ . It must be noted that in the SAR geometry shown in Figure 2.2 of Chapter 2, the points at which the signals are transmitted and received are equally spaced. This would then correspond to non-uniform angular increments. However, the CBP algorithm does not require uniform angular increments. Only the difference in the successive angles  $\Delta\theta$  is needed for computation. The smaller the value of  $\Delta\theta$ , the more accurate is the computation in the Back-projection step.

### 3.2.1 CBP Algorithm as Projectional Reconstruction

Figure 3.1 graphically illustrates the two-step CBP process. Each projection  $\{P_{\theta_k}(\omega_l)\}$  is individually multiplied by the Jacobian,  $|\omega|$ , shifted by the carrier frequency  $\omega_c$ , and then 1-D inverse Fourier transformed via an FFT. A transformed projection is then rotated by the angle of projection  $\theta_k$  and then the sample values are assigned to all the pixels of the image along the equidistant parallel lines, perpendicular to the projections. For the pixels that do not fall along these lines, 1-D interpolation of the projection is required. This two step operation is performed for all projections. Computation due to one projection results in a crude image. Contribution of the successive projections are added to produce the final high resolution image.

### 3.2.2 CBP Algorithm as a Point-by-Point Reconstruction

The actual implementation of SAR reconstruction is performed as a point-by-point reconstruction. It can be viewed as computing the values of the predefined image pixels. After shifting and 1-D Fourier transforming each projection within the limited angle, the projections would describe an hour glass in the projectional plane  $(t, \theta)$ . Given a pixel,  $(x_0, y_0)$  or  $(r_0, \phi_0)$ , there is an associated trajectory defined by  $t_0(\theta) = r_0(\cos(\phi_0 - \theta))$ . Figure 3.2 shows these trajectorial arcs for sixteen pixels. The integration operation as in equation 3.7 is performed along these arcs. Since the projectional data are discrete, the sample values of each projection need to



FD-8796

Figure 3.1: The CBP algorithm for SAR.

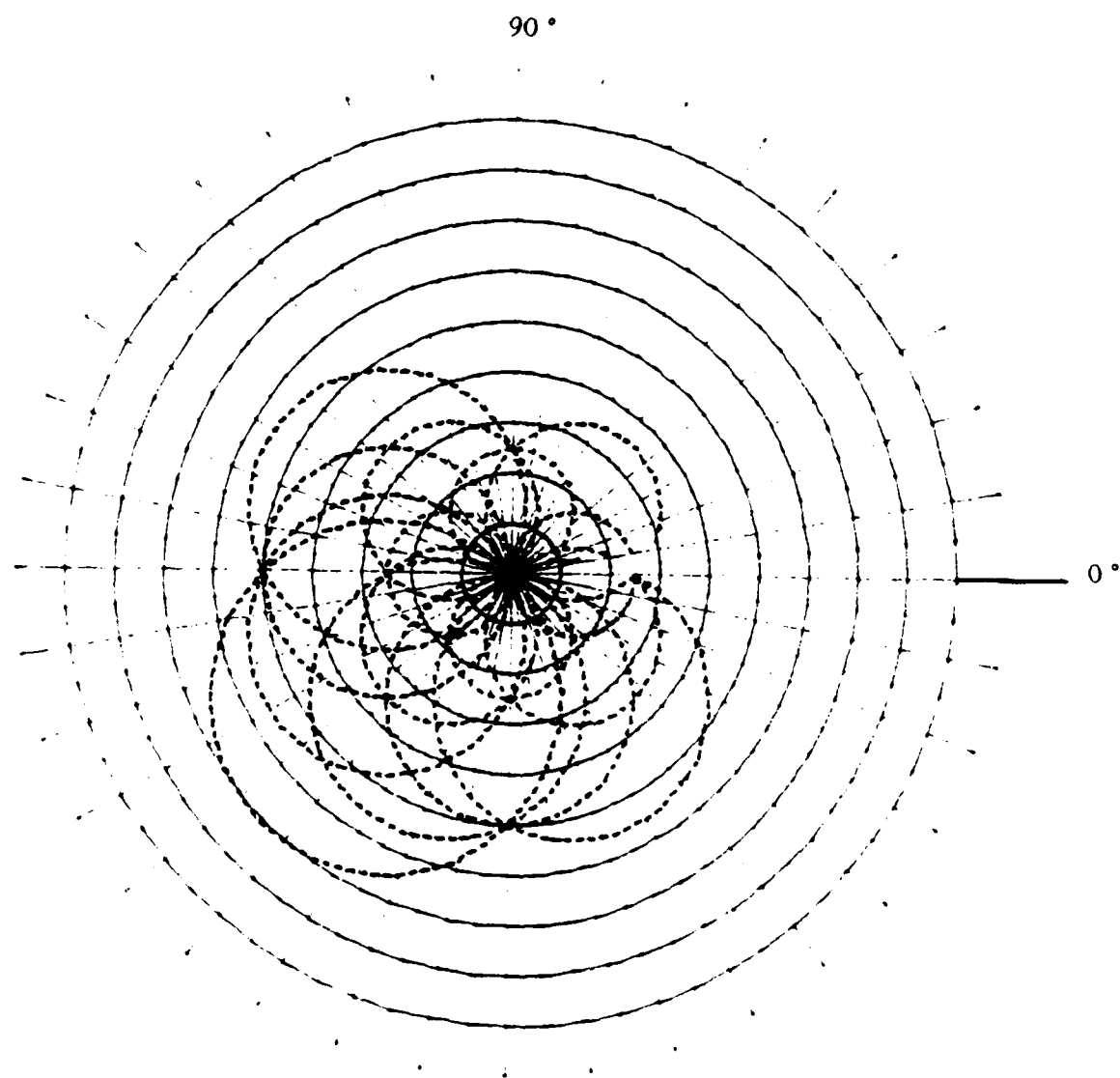


Figure 3.2: The trajectories of image pixels.

be interpolated to obtain the values of the points not corresponding to the sample points, before the integration can be performed. Any method of approximation can be used for integration. Therefore, the final pixel value is the sum of these interpolated quantities along the arc associated with that pixel.

### 3.3 Signal Processing Aspects of the Algorithm

#### Windowing :

Windows are applied to the Fourier domain data prior to transformation to reduce sidelobes and control the shape of the ideal point target response of the system. Windowing can be done in two ways :

- a). By applying a 2-D separable window  $W(\omega, \theta)$  in the Fourier space prior to 1-D FFT operation.
- b). By applying a window  $W_1(\omega)$  in the radial direction in the fourier space prior to 1-D FFT, so that the function to be interpolated becomes :  $P_\theta(\omega) |\omega| W_1(\omega)$ ; and then another windowed  $W_2(t)$  in the projectional space defined by the variables  $(t, \theta)$ , along the pixel trajectory prior to summation in which case the function to be integrated is  $Q(t) W_2(t)$ .

These windows,  $W_1(\omega)$  and  $W_2(t)$  can be any one of the existing windows like Hanning or Hamming.

#### Interpolation :

Since the samples of each projection are available, 1-D interpolation is required for each pixel in the back-projection step of Equation 3.10. The sampled values of  $Q(t)$  are known only at the points  $\{t_m : m = 1, 2, \dots, N_s\}$  and therefore it is necessary to compute the values at each  $t = r \cos(\phi - \theta_i)$ . Any of the conventional interpolators such as Nearest Neighbor, Linear, or relevant FIR filter functions can be used.

### 3.4 Complexity of the Algorithm

In order to determine the complexity of the algorithm, the flow of computation in the CBP algorithm has to be considered. This flow model has been represented using the following rather informal high-level description language.

#### Convolution Back-Projection

**Input :** A set of  $N_p$  projections  $\{P_1, P_2, \dots, P_{N_p}\}$ ,  
where each projection  $P_k$  is a sequence of  $N_s$  samples.

**Output:** A  $N_x \times N_y$  array  $g(i, j)$  of image intensities.

```

for each  $k = 1..N_p$  do
  begin
     $\hat{Q}_k = 1\text{-D inverse FFT of } P_k$ 
    for each  $i = 1..N_x$  do
      begin
        for each  $j = 1..N_y$  do
          begin
             $t = x_i \cos \theta_k + y_j \sin \theta_k$ 
             $\tilde{g}(i, j) = \tilde{g}(i, j) + \hat{Q}_k(t) * \text{phase} * \text{wgt} * \Delta \theta$ 
          end
        end
      end
    end
  end
end

```

For each projection, the algorithm first computes the 1-D FFT on the sequence of length  $N_s$ . This clearly requires  $O(N_s \log_2(N_s))$  computations. In addition, there are  $O(N^2)$  operations within the two inner loops. Therefore, the total number of operations required is  $O\left[N_p \times [N_s \log(N_s) + N_x N_y]\right]$  which is simply  $O(N_p N_x N_y)$ , provided  $N_s$  has the same order of magnitude as  $N_x$  and  $N_y$ . Furthermore, if  $N_p = N_x = N_y = N$ , then clearly the complexity of the CBP algorithm reduces to  $O(N^3)$ .

## CHAPTER 4

### ARCHITECTURE FOR SAR IMAGE RECONSTRUCTION USING CBP

One major advantage of Convolution Back-Projection algorithm is the fact that it contains inherent parallelism. This important characteristic can be exploited by using linear systolic array architectures to reduce the computation time. To utilize the inherent parallelism of the CBP algorithm to its fullest extent, it is essential to design an appropriate special purpose architecture. The availability of low-cost VLSI chips with large numbers of gates has made the implementation of special purpose architectures feasible. The cost of development of such hardware can be justified by an expected increase in throughput rate. In this chapter, architectural implications of the CBP algorithm will be discussed and an architecture for a SAR processor is suggested.

The CBP algorithm offers certain features that appear to be well-suited for VLSI implementation, and can lead to real-time processing. The convolution stage of the CBP algorithm can be implemented immediately as the spectral slices (or projections) are sampled and stored. Further processing can be done as well, before the next spectral slice is available. For example, immediately after each projection is filtered, 1-D interpolation can be immediately started, and the appropriate samples from this projection can be added into proper pixels in the final image, as called for by the back-projections, so that the final image can be produced shortly after the final projection has arrived.

#### 4.1 Nature of Computation

It was shown in the last chapter that SAR is a point-by-point reconstruction algorithm. Based on this fact, the nature of computation required will be considered here. Let the reconstructed image be  $\hat{g}(x, y)$  be an array of  $N \times N$  pixels. Also, let the set of transformed projections be denoted by

$$\left\{ Q^k(l) : \begin{array}{l} l = 1, \dots, N_p \\ k = 1, \dots, N_s \end{array} \right\}$$

where  $N_p$  is the number of projections and  $N_s$  is samples per projection. For each projection, and for each pixel in the image, i.e., each  $x_i$  and  $y_j$ , the intensity value must be computed. Note that given a projection, the computation necessary for each pixel  $(x_i, y_j)$  is independent of any other pixel of the image. This is valid for all projections. The value for a pixel is dependent only on the input, the previous and present intensity values

The computation required for each pixel  $(x_i, y_j)$  or  $(r, \phi)$  given a projection at an angle  $\theta^k$  can be divided into the following steps:

- a) Computation of the position at which the projection value is desired, i.e., computation of  $t_{i,j}$  for each pixel  $(r, \phi)$ :

$$t_{i,j} = r \cos(\phi - \theta^k)$$

- b)

- i) Access of the two projectional data values indexed by the two integers enclosing  $t_{i,j}$ :

$$Q^k(l) = \left\lfloor Q^k \left\lfloor t_{i,j} \right\rfloor \right\rfloor \quad \text{and} \quad Q^k(l+1) = Q^k \left\lceil \left\lceil t_{i,j} \right\rceil \right\rceil$$

- ii) Access of the constant  $d\theta$ .

- c)

- i) Computation of  $K(\omega_c, t_{i,j}) = e^{j\omega_c t_{i,j}}$
- ii) 1-D interpolation between  $Q^k(l)$  and  $Q^k(l+1)$  to obtain  $Q^k(t_{i,j})$ .
- iii) Computation of the contribution of this projection.

$$\hat{g}^k = Q^k(t_{i,j}) * K(\omega_c, t) * d\theta$$

- d). Addition of the present contribution to the previous contributions and storage of the new value in the local memory.

$$\tilde{g}^i \leftarrow \tilde{g}^{i-1} + \hat{g}^i$$

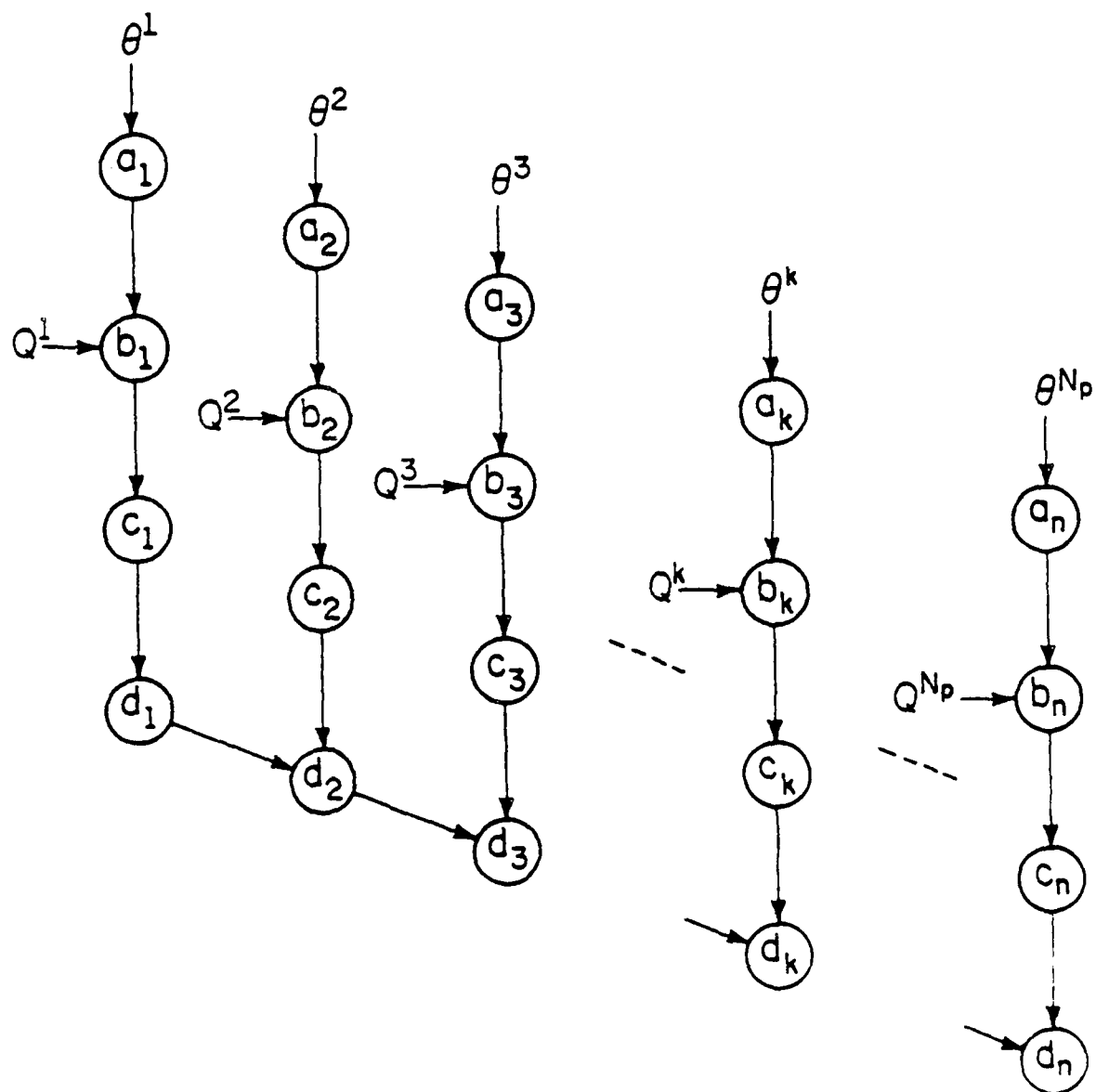
Consider the computation required for one pixel. The data dependency among these four steps for all  $N_p$  projections has been graphically represented in Figure 4.1. Each node in the figure represents a computational step, and the subscripts  $i$  ( $i = 1, 2, \dots, N_p$ ) stand for the projection number. An arrow incident from Node  $a$  onto Node  $b$  indicates that Step  $b$  requires the results of Step  $a$  for its evaluation; so, Step  $b$  cannot be started until Step  $a$  has been completed.

## 4.2 Architectural Considerations

### 4.2.1 One Processor Per Pixel

Consider one processor per pixel, each one performing the required  $a$ ) to  $d$ ) steps outlined in the previous section. Many processors may be required to access the same location of the projectional array at the same time. Since this is not possible with the requirements of maintaining high speed and simplicity of logic, the projectional array must be read into a local memory assigned to each processor. The angle  $\theta^i$  and the projectional values  $\{Q^i(l)\}$  are loaded once before the beginning of the computation in the local memory. The computation can be performed in a pipelined fashion, with separate Stages  $A$  to  $D$  for each step  $a$ ) to  $d$ ) respectively. Figure 4.2 shows the block diagram of the individual processor. Data flow from one stage to the next. Finally, the result from the Stage  $D$  is stored in local memory. During the computation for the next projection, this old value has to be brought in to Stage  $D$  and added to the present value to update its value in the local memory.

This scheme of one processor per pixel has the tremendous advantage of achieving  $O(N_p)$  time-complexity. However, the number of processors required ( $N \times N$ ) may be as large as  $10^6$  for a  $1K \times 1K$  image, which is impractical. These processors will be under-utilized, since the computation at each processor can be performed at a much faster speed than the input data rate



FP-8804

Figure 4.1 The data dependency diagram.

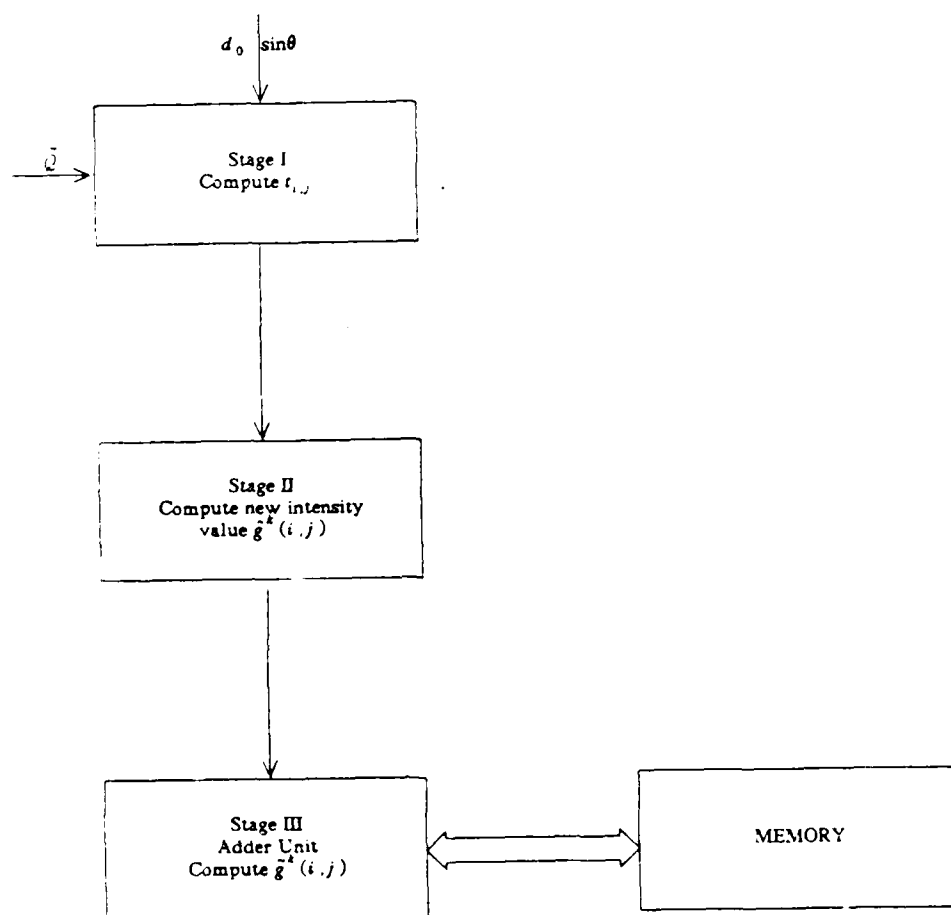


Figure 4.2 The Block diagram for each processor ( processor per pixel ).

Each processor requires a large local memory.  $N^2$  copies of such a local memory is too expensive.

#### 4.2.2 One Processor Per Column

An efficient and cost-effective approach is to use one processor on each column of the image; thereby requiring only  $N$  processors. As the new set of data arrives, each processor copies its relevant portion of the data into its local memory. Let  $P^k_{i,j}$  denote the processing required for the pixel of the  $i$ th row and the  $j$ th column. The super-script  $k$  represents the processing that is associated with the  $k$ th set of projections. Each of the processors sequentially computes the values of the  $N$  pixels in its column. This scheme of one processor per column has the merit of reducing the number of processors to  $N$ .

To further increase the efficiency of the architecture, the computation described in Steps a) to d) needs to be reorganized. First, this involves interchanging the order of the interpolation Step ( c (ii) ) and the multiplication Step ( c (iii) ). Secondly, it will be shown that the number of data values accessed by a processor is a small fraction of the total number of data items, thereby reducing memory requirements. Third, the costly step of calculating the  $t$  value of each pixel for each projection can be simplified.

The multiplication by  $K(\omega_c, t)$  and  $d\theta$  can be performed on all  $Q^k(l)$  values before these values are accessed by the processors. We will refer to these premultiplied values as  $\tilde{Q}^k(l)$ . Then  $\tilde{g}^k$  is obtained by interpolating between  $\tilde{Q}^k(l)$  and  $\tilde{Q}^k(l+1)$ . In doing so, the number of multiplication-by-constant steps is reduced from  $N^2$  to  $N_c$ . Since  $N_c$  is usually not more than  $3N$ , this results in substantial savings in the total computation required by a processor.

The number of sample points required by each processor is a function of only the projection angle  $\theta$  given by  $d(\theta) = N \sin(\theta)$ . Given a maximum value of  $\theta = \frac{\theta_m}{2}$ . The maximum number of projectional values, denoted by  $d$ , required per column is:

$$d = N \sin\left|\frac{\theta_m}{2}\right| \quad (4.1)$$

Note that this value remains constant for all  $N$  columns. As a result, only a small fraction of data need to be accessed by each processor. This will reduce the bandwidth of the data path to each processor by a factor of  $\frac{N_s}{d}$ . This results in substantial savings, since data paths account for a large portion of the total cost of the multiprocessor system.

The calculation of  $t$  as described in Step a) is computationally intensive. The following is a cost-effective procedure for computing  $t$  values for each pixel. See Figure 4.3.

Let  $d_0(j, \theta)$  be the location of the data value accessed by the center pixel of column  $j$ . It can easily be shown that,

$$d_0(j, \theta) = N_s/2 - (N/2 - j) \cos|\theta| \quad (4.2)$$

For a fixed  $j$ ,  $d_0$  ranges from

$$N_s/2 - (N/2 - j) \cos|\theta_m/2| \text{ to } N_s/2 + (N/2 - j) \cos(0)$$

for  $-\theta_m/2$  to 0. This range will remain the same for 0 to  $\theta_m/2$ , since  $\theta$  is symmetrical around 0. The total deviation for first and last columns will be

$$|\cos|\theta_m/2| - \cos(0)| * N/2$$

Usually,  $\theta$  is small for SAR applications, and therefore this deviation will be very small. In essence, the value accessed by a center pixel of a column,  $d_0$ , will remain more or less constant over all  $\theta$ . Therefore, the  $d_0$  need not be computed for each projectional angle. Assuming that  $d_0$  remains constant, the value accessed by the other pixels in the column can be determined from the deviation from the center pixel of the column. See Figure 4.4. For a pixel  $(i, j)$ , then  $t_{i,j}$  will be

$$t_{i,j} = d_0(j, \theta) + (N/2 - i) \sin\theta \quad (4.3)$$

and therefore, the contribution by the projection  $k$ ,  $\hat{g}^k$  is given by

$$\hat{g}^k(i, j) = \tilde{Q}^k(\lfloor t_{i,j} + 0.5 \rfloor) \quad (4.4)$$

The above formula assumes nearest neighbor interpolation.

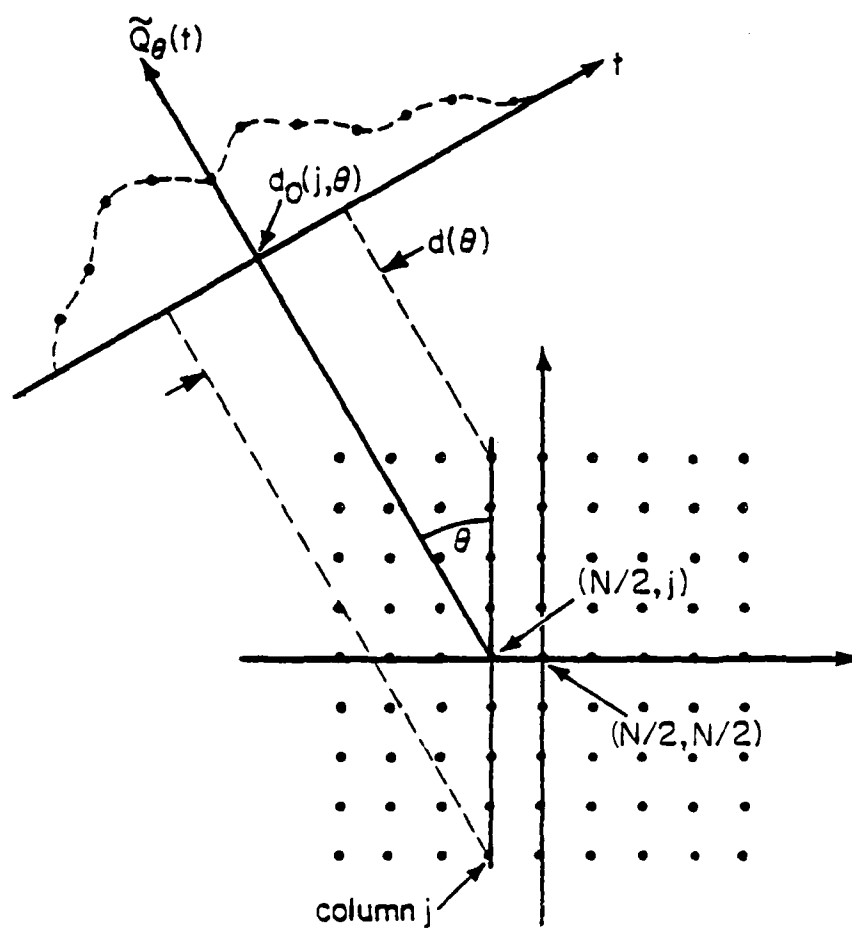
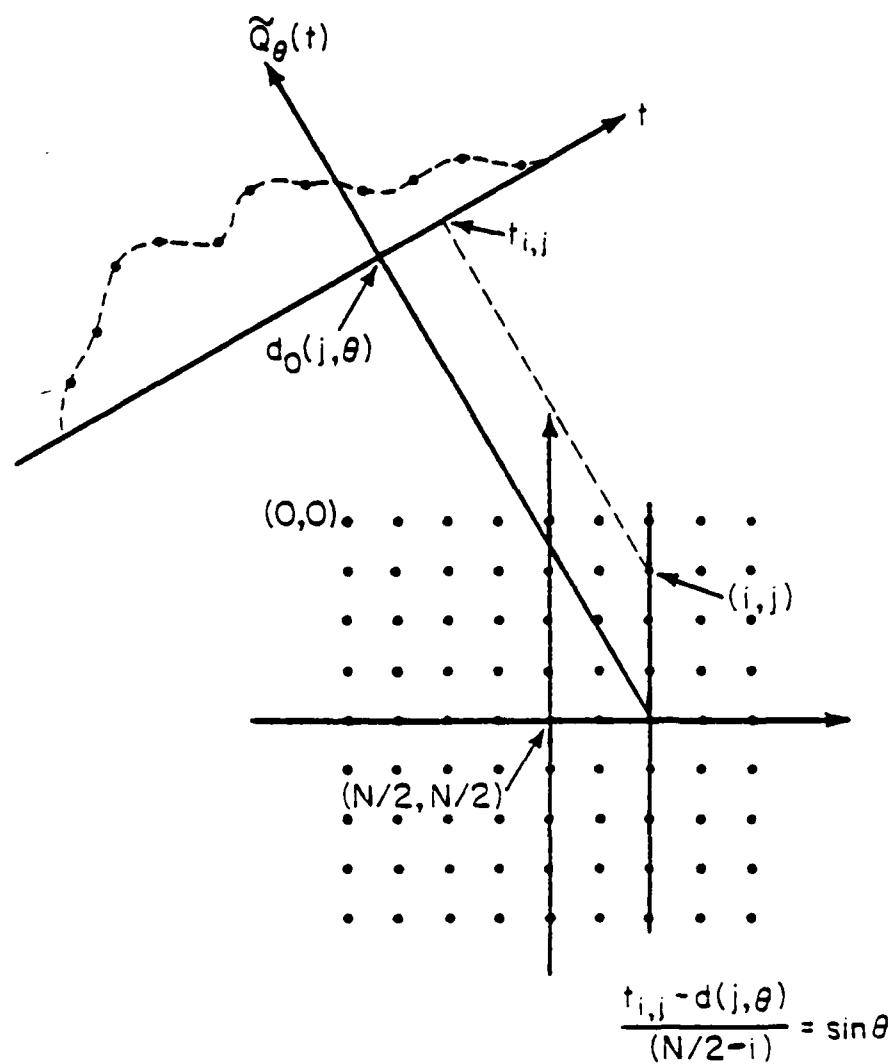


Figure 4.3 The location of  $d_0$  for each column.

Figure 4.4 Computation of  $t_{i,j}$ .

For linear interpolation, the equation is

$$\hat{g}^k(i, j) = \tilde{Q}^k(\lfloor t_{i,j} \rfloor) + (\tilde{Q}^k(\lfloor t_{i,j} \rfloor) - \tilde{Q}^k(\lfloor t_{i,j} \rfloor - 1)) \times (t_{i,j} - \lfloor t_{i,j} \rfloor) \quad (4.5)$$

Intensity value for the pixel  $(i, j)$  is given by

$$\tilde{g}^k(i, j) = \tilde{g}^{k-1}(i, j) + \hat{g}^k(i, j) \quad (4.6)$$

Thus the following procedure can be used to compute  $\tilde{g}^k$  from  $\tilde{Q}^k$ .

- 1) Start with the central pixel of the image,  $(i = N/2, j = N/2)$ , since  $t$  values for all angles will be zero. From the central pixel, find the  $t$  value for pixels  $(N/2, j)$ .
- 2) Compute  $d_0$  for the central pixel of the column  $j$ .
- 3) Once  $d_0$  is known, using Equation 4.4, compute  $t_{i,j}$  for all pixels of the column  $j$ .
- 4) Finally, using Equation 4.5, the total value of a pixel can be calculated and stored in the local memory.

The above procedure is repeated for each projection.

The block diagram for a single processor is shown in Figure 4.5, where each stage performs the required processing as described below.

- I. Computing  $t_{i,j}$  for each pixel in the column using Eq. 4.3.
- II. Computing the new intensity value,  $\hat{g}$  as in Eq. 4.4.
- III. Updating the memory by adding this new value to obtain  $\tilde{g}$ .

Since computation in III is dependent on II, and it in turn dependent on I, the computation can be done in a pipelined fashion for each pixel. The reservation table [16,33] of Figure 4.6 describes the timing of the pipeline. Once again,  $P^k_{i,j}$  denotes the processing required for the pixel of  $i$ th row and  $j$ th column and the superscript  $k$  denotes the processing associated with the  $k$ th projection. Each column of the reservation table denotes the time step required. Each row denotes the computation performed in the corresponding processor stage. Let  $T_p$  be the time between two consecutive projections. Since each step has to be executed for each pixel in

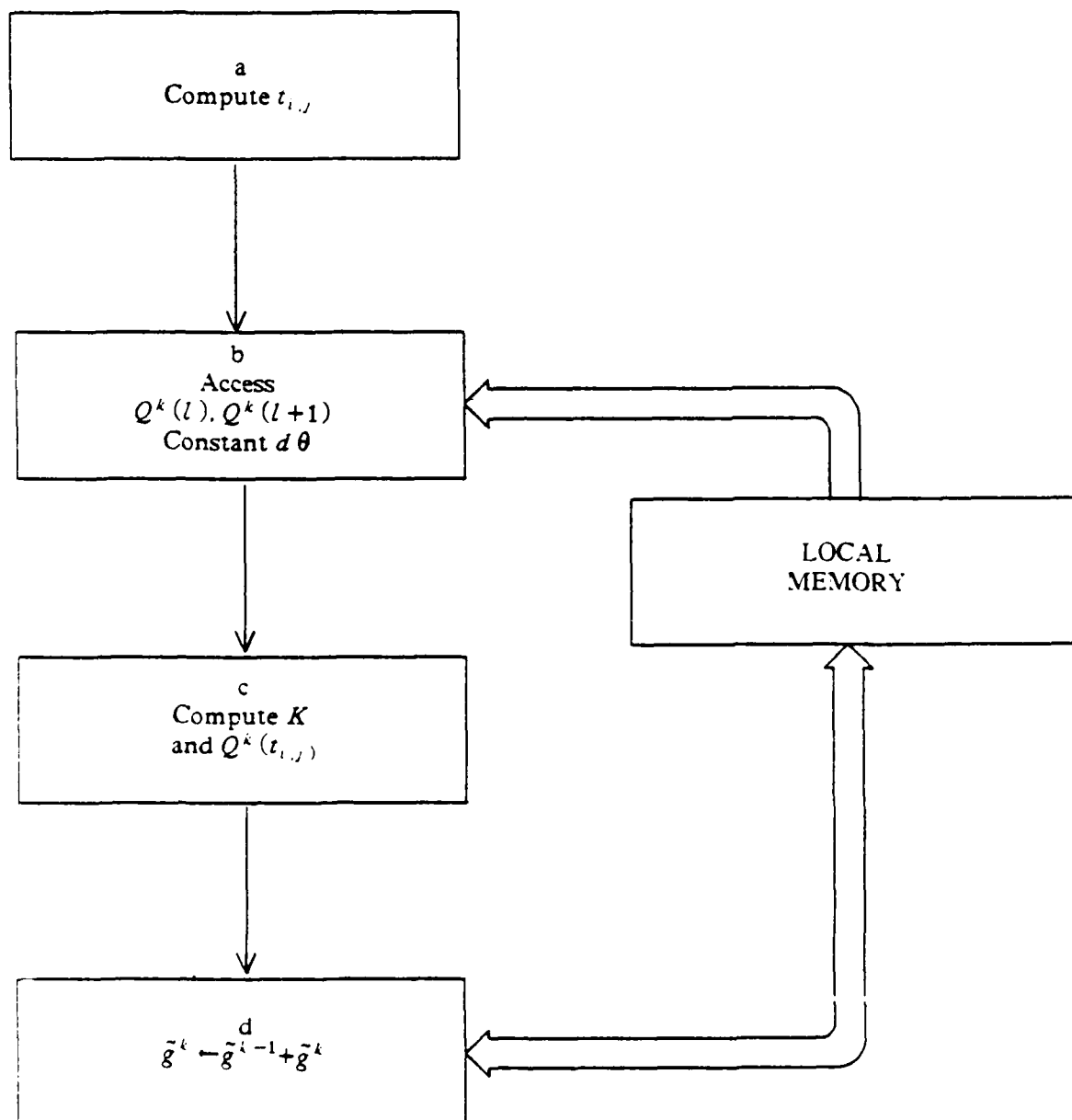


Figure 4.5 The block diagram for the individual processor for each column.

	Time								
	1	2	3	4	N-1	N	N+1	N+2	
Stage I	$P_{1,j}^I$	$P_{2,j}^I$	$P_{3,j}^I$		$P_{n-1,j}^I$	$P_{n,j}^I$			
Stage II		$P_{1,j}^I$	$P_{2,j}^I$		$P_{n-2,j}^I$	$P_{n-1,j}^I$	$P_{n,j}^I$		
Stage III			$P_{1,j}^I$		$P_{n-3,j}^I$	$P_{n-2,j}^I$	$P_{n-1,j}^I$	$P_{n,j}^I$	

$T_p$  : Time between projectional arrays

Figure 4.6 The reservation table for column  $j$ .

the column and since there are  $N$  of them in a column, each step takes at most  $\frac{T_p}{N}$  time periods. Two more time steps will be required to complete the image from one projection, however, the processing for the next projection can resume. Therefore, the final image will be produced two time periods after the last projection is received.

### 4.3 Feasibility of the Architecture

Figure 4.7 shows the block diagram of the architecture required. It is a Linear Systolic Array (LSA) with  $N$  identical processors linearly connected with both input and output clocked from left to right. Given  $\theta$ , the multiplier multiplies each  $Q$  by  $K(\omega, t)$  and  $d\theta$ . The output of the multiplier is denoted by  $\tilde{Q}(t)$ . The  $N_s$  values of  $\tilde{Q}(t)$  are clocked in at the left most cell every  $T_s = \frac{T_p}{N_s}$  seconds. It is necessary that the data flow from one cell to the next at a rate of  $T_c$  seconds for such an LSA structure.

Assume that the data rate, i.e., the projection rate is  $T_p$  and that there are  $4K$  projections and  $2K$  samples per projections. Also assume that  $-1.5^\circ \leq \theta \leq 1.5^\circ$ . With these data an image of the size  $1K \times 1K$  pixels has to be reconstructed. Let  $C$  be the number of operations required for one pixel per projection; thus, the total amount of computation required for one pixel is of the order  $4K\{C\}$ . One processor per column would require the processor to perform  $1K\{C \times 4K\}$  operations in  $T_p$  seconds. With such requirements, an outline of a generic processor structure is presented here.

As shown in Figure 4.7, the digital projection values  $P_d(\omega)$  are the input to the 1-D FFT processor. The output,  $Q_d(t)$  is then passed through a multiplier which multiplies each sample by  $d\theta$  and  $K(\omega, t)$ . Then  $N_s$  of these  $\tilde{Q}_d(t)$  values are clocked in to the LSA. As the data flow through each cell,  $C_j$ , they obtain  $d$  values centered at  $d_{j0}$ . For our example,

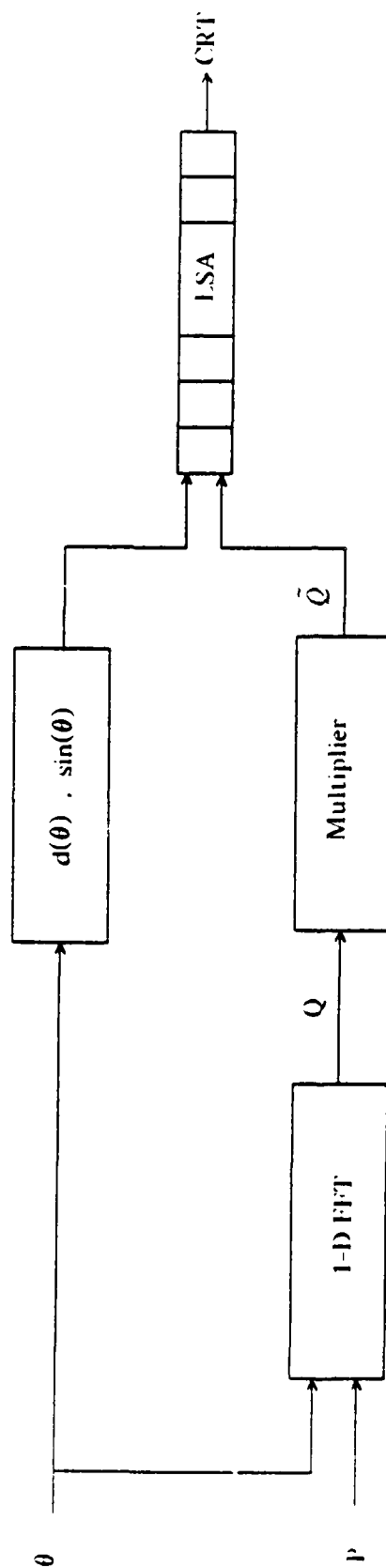


Figure 4.7 The block diagram of the required architecture.

$d = 1000 \times \sin(-1.5^\circ) = 27$  memory words. The amount of local memory required per processor is extremely small. After the data have been obtained, the cell  $C_j$  computes the updated intensity values for each pixel in the column. Thus, an image from one projection is created. Since there is no need to display the image after every projection, the pixel intensity values can be clocked out from left to right after every ten projections. From the right-most cell, the output is sent to a display device.

The proposed architecture is realizable with present day VLSI technology. The input data rate,  $\frac{N_s}{T_p} = 2\text{M}$  bytes per second, and the output data rate of  $\frac{1K \times 1K}{10T_p} = 100\text{M}$  bytes per second is very realistic. As can be seen from Steps I. through III., the number of operations required per pixel is reasonable, and therefore, the throughput of each processor can meet the required rate.

Since the back-projection stage is a point-by-point reconstruction process, each column could be computed in a separate processor, each of which carries out the back-projection for all pixels in the column. Each processor is simple, consisting mainly of a complex-number multiplier, adder, and accumulation register. This suggests a systolic array structure, i.e., a structure that consists of a large number of similar cells which require a minimum of intercell communication. As an alternative to providing a processor in each column, the image can be segmented into small subarrays, with one processor implementing the back-projection for all pixels in a particular subarray.

Both of these architectural alternatives also provide some degree of fault tolerance. If a processor fails, a column or a subarray is lost, but the other processors will continue to produce the rest of the image correctly. Single error detection capabilities can be added to the processors by using Quadratic Modular Number (QMN) codes [35,25,26] with one redundant moduli to perform the arithmetic. For multiple error detection, additional redundant moduli are required. With the addition of  $r$  redundant moduli, it is possible to detect  $r$  errors or correct

$n/2$  errors. This scheme would involve some overhead in transforming to and from the QMN representation. Quadratic modular number codes decompose a complex multiply into two parallel real multiplies, thereby simplifying complex multiplication and providing error isolation between the channels. After detecting the error, it is a relatively simple matter to identify a faulty processor, and then assign one of the remaining good processors to take over its function. Such a structure has the properties of soft failure in the sense that as processors fail, the overall throughput rate is damaged (or the image size is reduced), but the system is able to reassign its resources in order to continue functioning in a useful way.

## CHAPTER 5

### SIMULATION AND RESULTS

The results of Chapter 3 show that the convolution back projection algorithm is a theoretically viable algorithm for spotlight mode Synthetic Aperture Radar. In this chapter, a quantitative evaluation of the CBP algorithm for SAR is presented using computer simulation. The CBP Algorithm is tested first on an ideal set of data points and then on a simulated model of non-ideal returned data from a point target. The characteristics of CBP reconstruction discussed in Chapter 3 (such as the projectional reconstruction and point-by-point reconstruction) are illustrated by experimental examples. To further establish its validity and accuracy, its performance is compared with the conventional FFT based method with respect to multiplicative noise ratio (MNR) and central processing unit (CPU) time.

The comparison between the CBP and DF methods is not entirely valid due to the following reasons. The SAR system generates the Fourier data on an annular region and the CBP algorithm uses this data directly. The FFT based method, however, inscribes a rectangular region within the annular region. Hence, the two methods invert slightly different regions of the Fourier domain. In order to obtain the samples on the rectangular region, the FFT based method uses a 2-D interpolator on the annular data. Using such an interpolator, requires data outside the original annular region so that the interpolated data at points close to the borders of the rectangular region may be computed accurately. Furthermore, the higher the order of the interpolator used, the wider is the annular region required. Hence, the SAR system has to generate additional data over a wider annular region for the FFT based method than for the CBP method to obtain similar reconstructional quality.

### 5.1 Computer Simulation for Ideal Data

In this section, the ideal point targets are considered to determine the algorithmic characteristic of the CBP algorithm modified for SAR image reconstruction. A point target of unit magnitude is assumed to be located at  $(x_t, y_t)$  on a rectangular grid in the image plane. The image reflectivity function of such a target is then

$$g(x, y) = \delta(x - x_t, y - y_t) \quad (5.1)$$

Its Fourier transform,

$$G(\omega, \theta) = e^{j(\omega x_t \cos(\theta) + y_t \sin(\theta))} \quad (5.2)$$

is evaluated on a polar grid shown in Figure 5.1, where the width of the annular region is  $\Delta\omega$ , the range of angle is  $\theta_m$ , the angular increment is  $\Delta\theta$ , the number of samples per projection is  $N_s$  and the number of projections is  $N_p$ .

The region to be mapped is of the width  $W_x$  and  $W_y$  in the  $x$  and the  $y$  directions, respectively. For the simulation it is assumed that  $W_x = W_y = W$  meters. In order to reconstruct the image digitally, the above region is divided into  $(64 \times 64)$  pixels. Each of these pixels are then  $\frac{W}{64} = \delta$  meters apart.  $\delta$  is also referred to as a resolution cell width. The reference point of the region is the origin.

The Fourier transform of the ideal point target is non-zero over the entire Fourier domain. In SAR image reconstruction, however, only a piece of this Fourier transform is provided on the polar grid of Figure 5.1. Hence, the inverse Fourier transform of this piece would be a 2-D sinc-type function which would then be the reconstructed image. The mainlobe of such a point target response will be a region of width  $\frac{4\pi}{\Delta\omega}$ . The location of the point target may vary from example to example, however in most cases it is placed on the image plane at integer multiples of the resolution cell width  $\delta$  which is defined to be  $\frac{2\pi}{\Delta\omega}$ . Motivation for such a choice is to produce an image with all the grid points outside the mainlobe to fall on the nulls of the 2-D

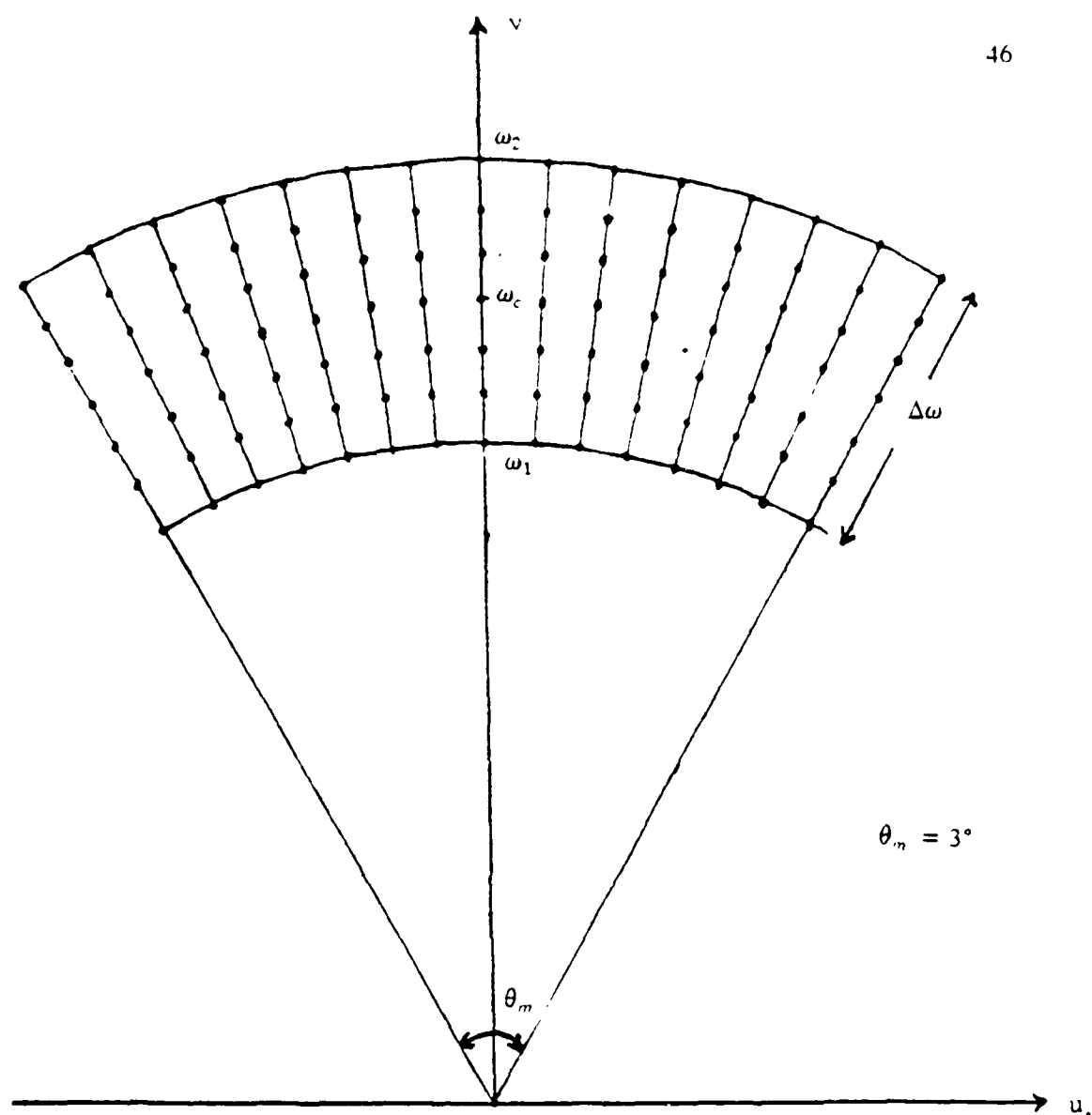


Figure 5.1 : Annular region of the Fourier space.

sinc function [55]. Thus, the sidelobes will not appear in the reconstructed image, even though they are present. Applying a Hamming type window on the Fourier data expands the mainlobe to twice the original width while suppressing the sidelobes considerably. Hence, the mainlobe of the windowed point target response will be  $5 \times 5$  pixels wide. So all peaks other than the mainlobe of the point target response can be designated to be independent of the data and only a function of the algorithm.

In an actual data collection situation, it is not possible to illuminate only the region to be mapped. In practice, the region illuminated by the antenna beamwidth is larger than the region to be mapped. This results in two sampling rates. One is associated with the region to be mapped and another with the illuminated region. Hence, the data sampling rate is greater than the Nyquist rate corresponding to the region to be mapped. In real situations, the data are 'over-sampled' to match the antenna beam-width and then filtered and resampled at a lower rate. The process of filtering and resampling is known as the pre-summing process. For the simulation purposes, pre-summing is not necessary. It is avoided by placing all the targets inside the region to be mapped.

The reconstructed image is displayed as a two-dimensional intensity function with a 60 dB floor. The point target response is evaluated by the multiplicative noise ratio (MNR) defined by

$$\text{MNR} = 10 \log \left| \frac{\sum (\text{pixels outside main lobe})^2}{\sum (\text{pixels inside main lobe})^2} \right|$$

The "mainlobe" is set to be a  $5 \times 5$  pixel square centered at the peak.

### 5.1.1 Illustration of Algorithmic Characteristics

#### Image Evolution

To illustrate the projectional interpretation of the modified convolution back-projection algorithm, the following experiment was performed. As a projection is computed, the two steps of convolution and back-projection as explained in Section 4.2.1 are performed on a projection to obtain an image due to this one projection. As additional projections are computed, the intensity contributions of these are added to the previous intensity values to update the image. Thus, one can perceive the *evolution* of the image. Figure 5.2 a) - h) shows the reconstruction of an ideal impulse from a number of projections, for  $N_p = 2, 10, 20, 30, 40, 50, 60$ , and 64. From only two projections, the range of the target is already determined. As the intensity contributions of more and more projections are added, the target slowly improves in the azimuth also. Figure 5.3 shows a plot of MNR versus the number of projections used. From this plot, it can be seen that the quality of the image i.e., the MNR, improves rapidly at the beginning, but the rate of improvement reduces considerably as number of projections becomes larger.

The practical implications of this image evolution process could be important in practice. For a reconnaissance aircraft, this evolution process can help recognize targets simultaneously with data collection. One need not wait for all the projectional measurements to be collected before an 'image' can be produced. This gives the option of prolonging the flight to ensure the target recognition or terminating it if the target has already been recognized. This dynamic aspect of the algorithm can be useful in surveillance applications as it provides an efficient on-line image reconstruction scheme which can generate a crude version of the enemy target very rapidly and improve upon the quality, if necessary, along the flight path. A similar an image evolution process is possible, if desired, with the conventional 2-D Fourier based method by setting the unknown Fourier data to zero and then performing the 2-D inverse FFT. The

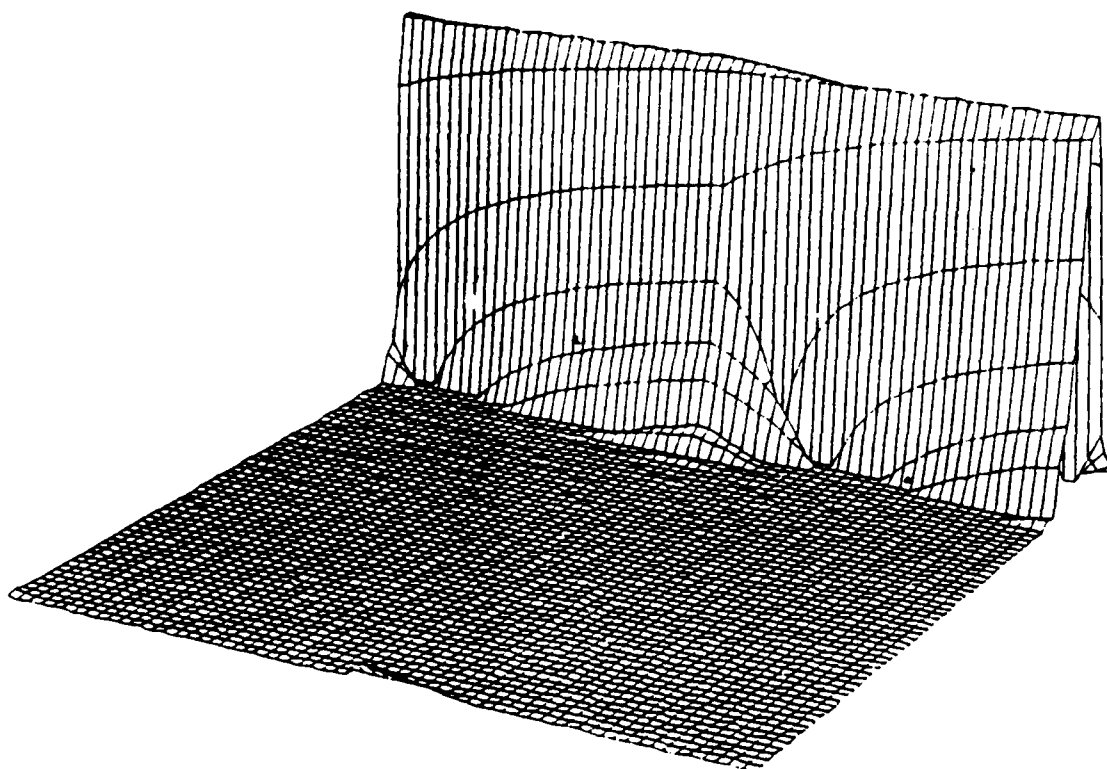


Figure 5.2 a) : Reconstruction with  $N_p$  (number of projections) = 2.

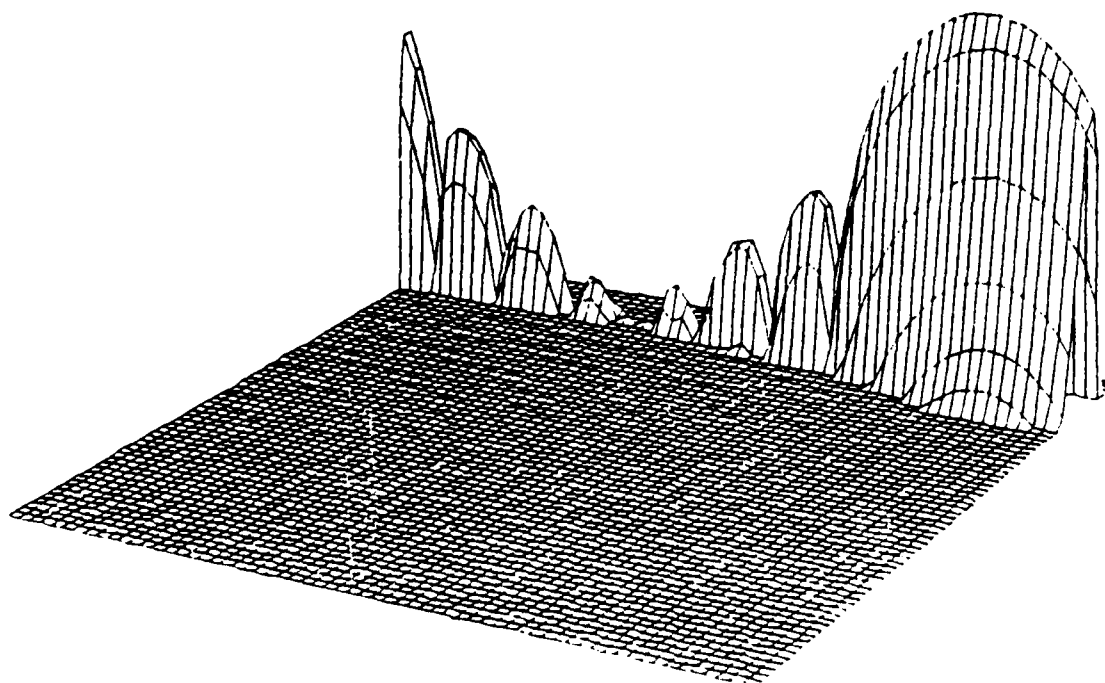


Figure 5.2 b) : Reconstruction with  $N_p$  (number of projections) = 10.

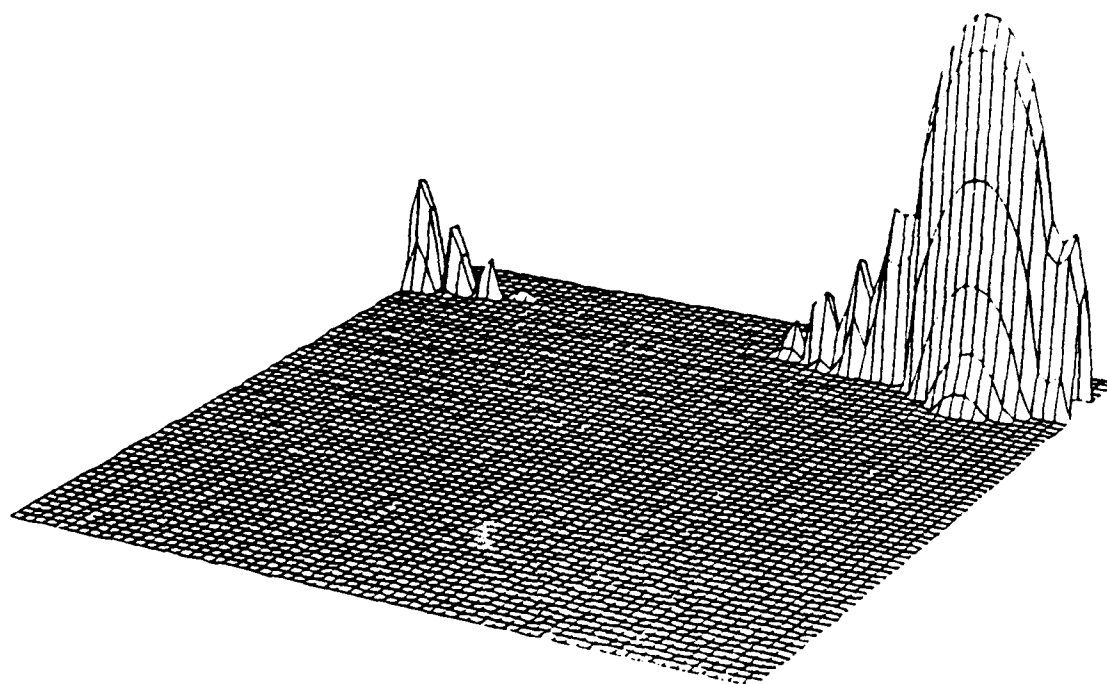


Figure 5.2 c) : Reconstruction with  $N_p$  (number of projections) = 20.

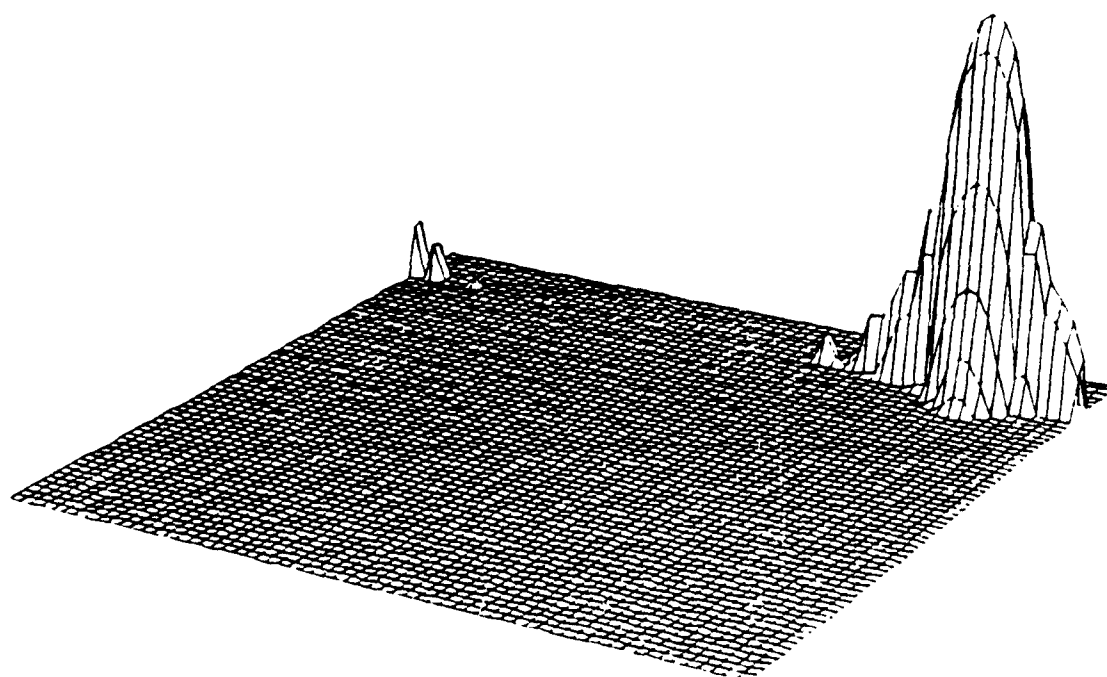


Figure 5.2 d) : Reconstruction with  $N_p$  (number of projections) = 30.

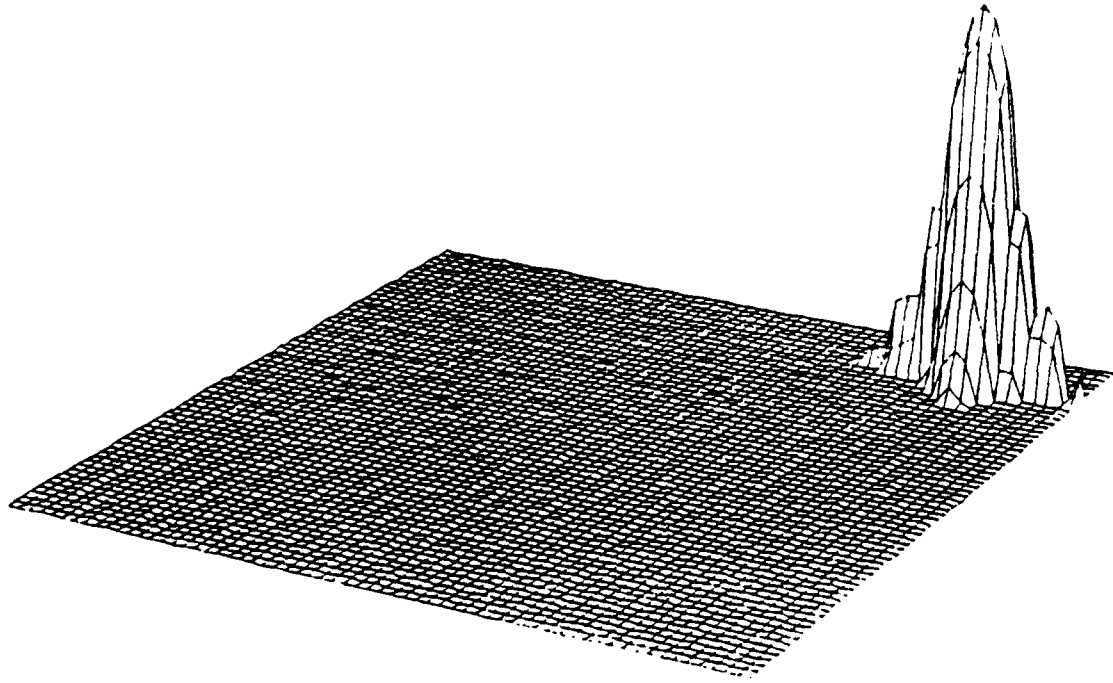


Figure 5.2 e) : Reconstruction with  $N_p$  (number of projections) = 40.

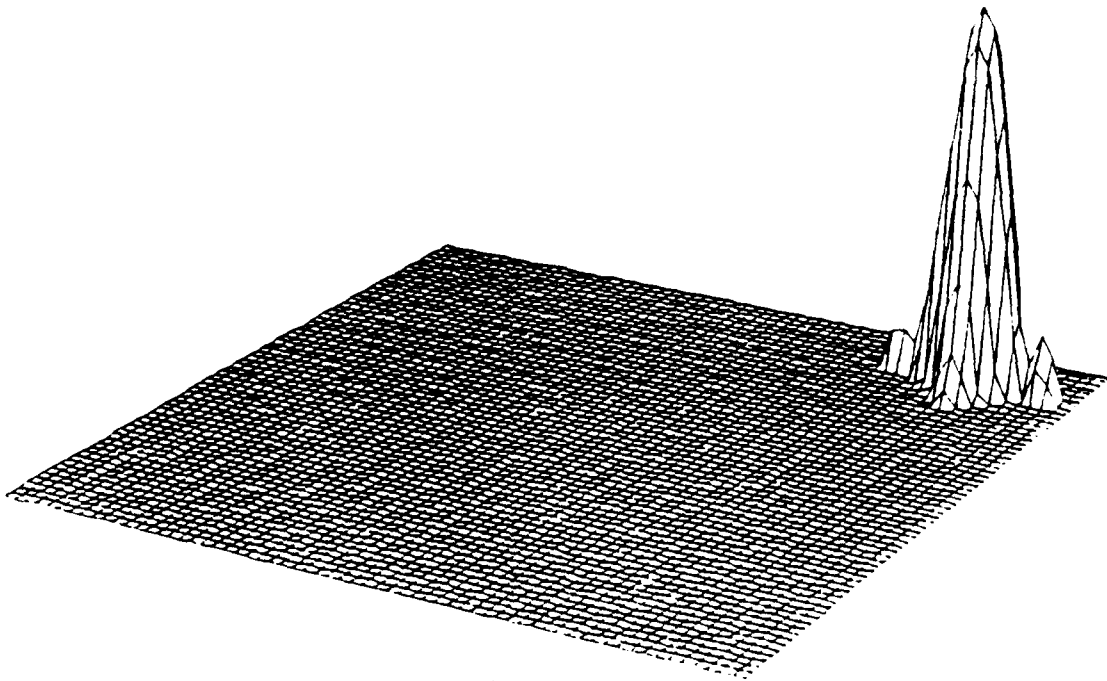


Figure 5.2 f) : Reconstruction with  $N_p$  (number of projections) = 50.

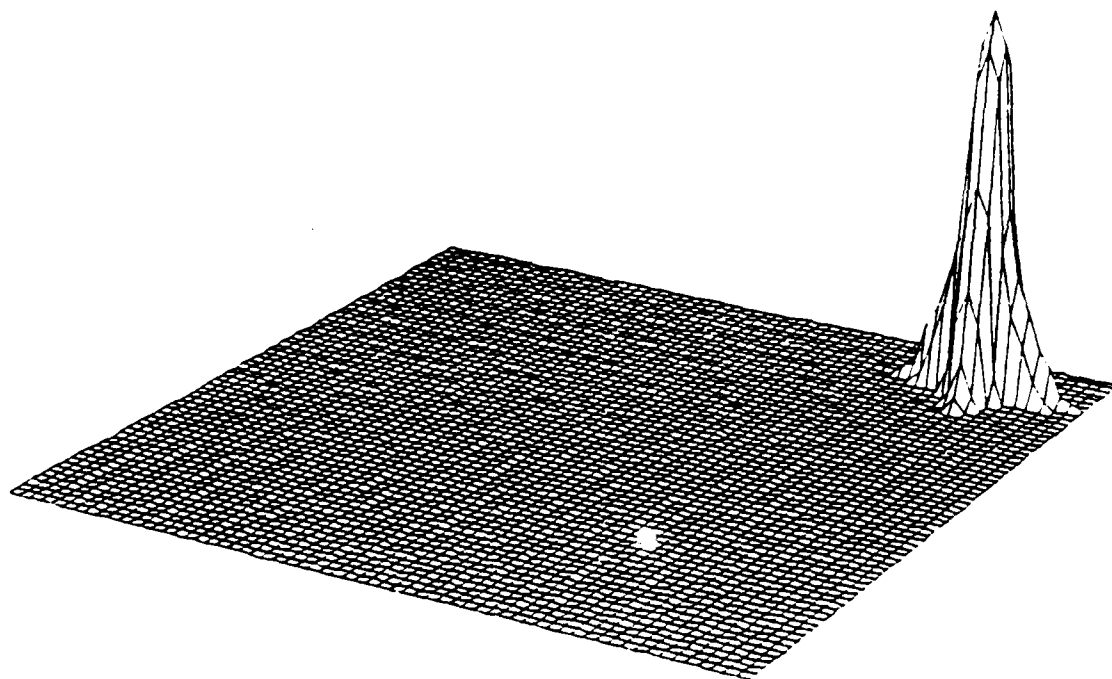


Figure 5.2 g) : Reconstruction with  $N_p$  (number of projections) = 60.

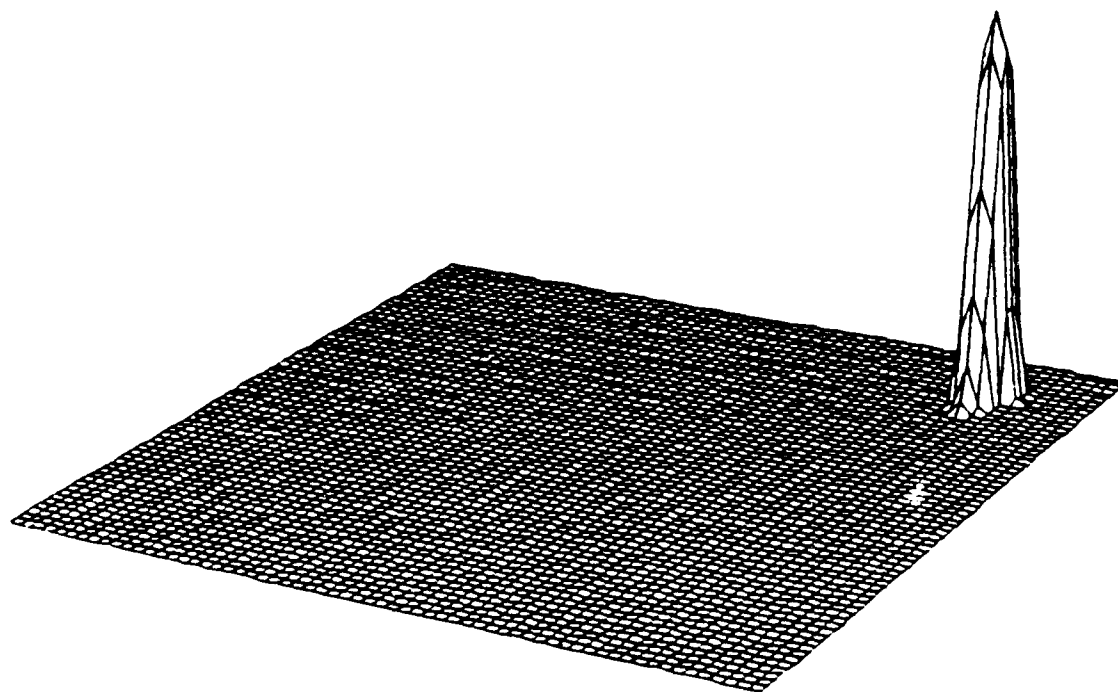


Figure 5.2 h) : Reconstruction with  $N_p$  (number of projections) = 64.

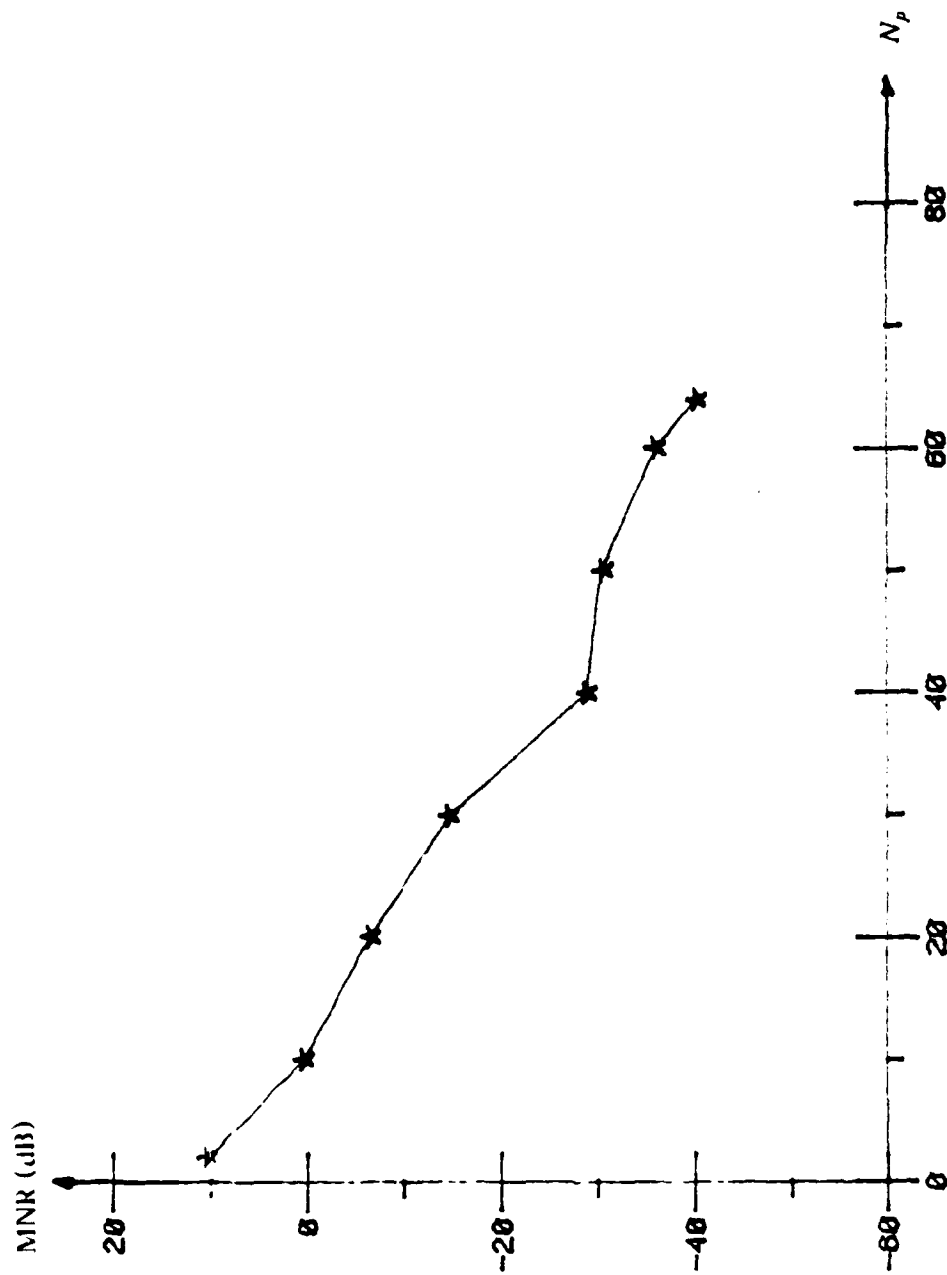


Figure 5.3 : MNR versus number of projections for subarray 16.1.

computation of the image intensities in this case, however, is not cumulative as in the CBP case.

### Reconstruction of Subimages

To illustrate the point-by-point reconstructional aspect of the algorithm, a subimage of  $(32 \times 32)$  pixels in the target field is considered. Figure 5.4 (a) is an example of the entire image of  $64 \times 64$  pixels. Figure 5.4 (b) shows the subimage of  $32 \times 32$  pixels centered at the origin. Figure 5.5 (a) is another example of a full image; Figure 5.5 (b) is the corresponding subimage of the lower right quadrant. In both these examples, all 64 projectional values were used to compute the subimages. As a projection is obtained, it is first 1-D inverse Fourier transformed, and then back-projected only for the points within the subregion of interest. Referring back to Equation 3.10, in Chapter 3, the choice of the points in the image plane at which the intensity values are desired are left entirely to the user. To illustrate this point, the region considered in Figure 5.4 (b) was divided into  $64 \times 64$  pixels instead of  $32 \times 32$  pixels as described above. The reconstructed image in this case is shown in Figure 5.6. This characteristic referred to as *image zooming* enables the user to examine the relevant parts of the image plane more closely and on a finer grid. In summary, on using the CBP algorithm, one can choose the points to be reconstructed, and then compute the intensity values for these points only.

The above characteristic of the CBP algorithm can be useful if the approximate target location is known *a priori* and also for the purpose of auto-focusing. If an approximate location of the target is sensed by some other means, such as infrared or laser sensors, then only the particular region around it need be reconstructed. In this situation, the algorithm can result in substantial savings in the computation time. Furthermore, this feature can also aid the auto-focusing step required for all SAR images [4]. Usually, an entire image is reconstructed and then certain subregions are correlated to determine the focusing parameters. Using this information, the data are then modified and the focused image is reproduced again. Using the CBP method, only the subregions necessary for auto-focusing purposes need be reconstructed, thus

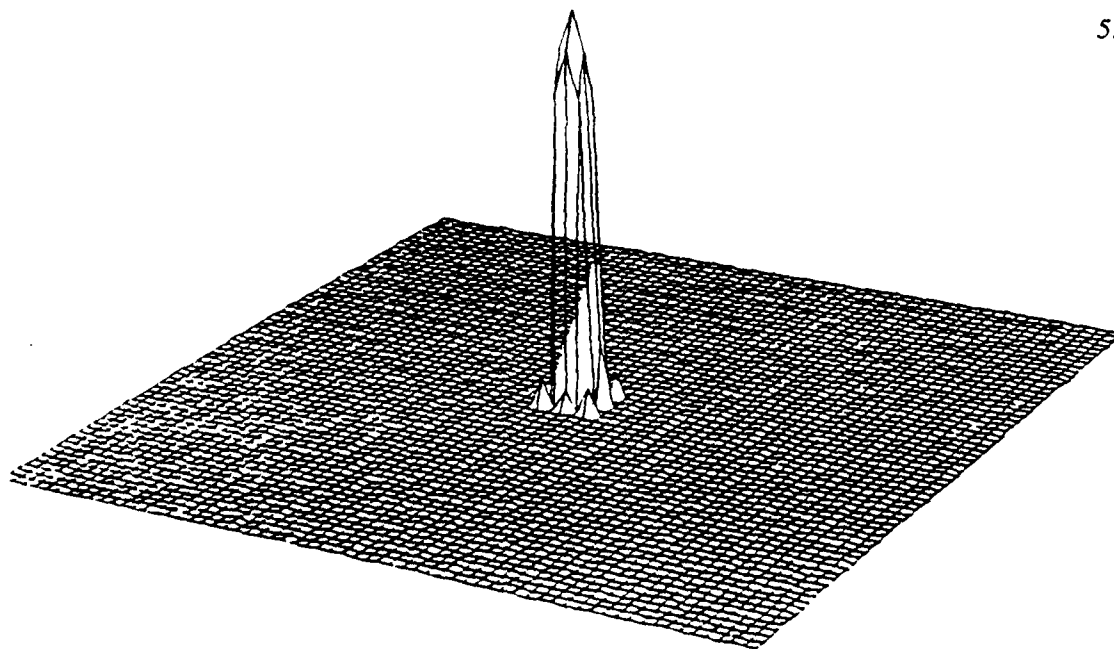


Figure 5.4 a) : Point target at (0..0.).

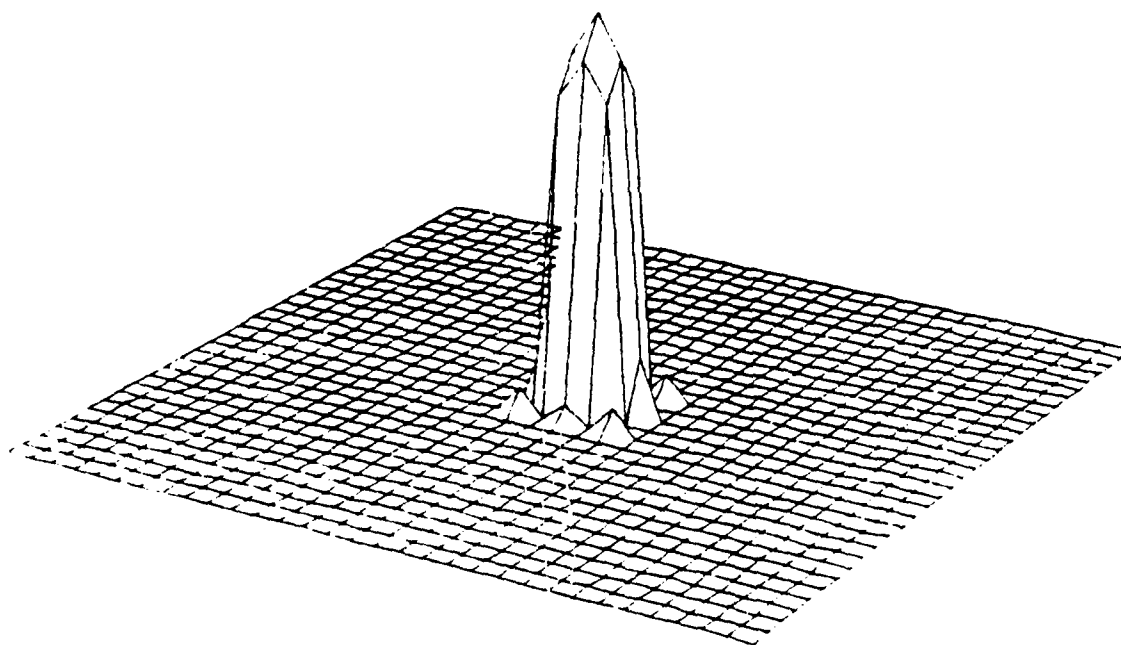


Figure 5.4 b) : Subimage of Figure 5.4 a) centered at origin.

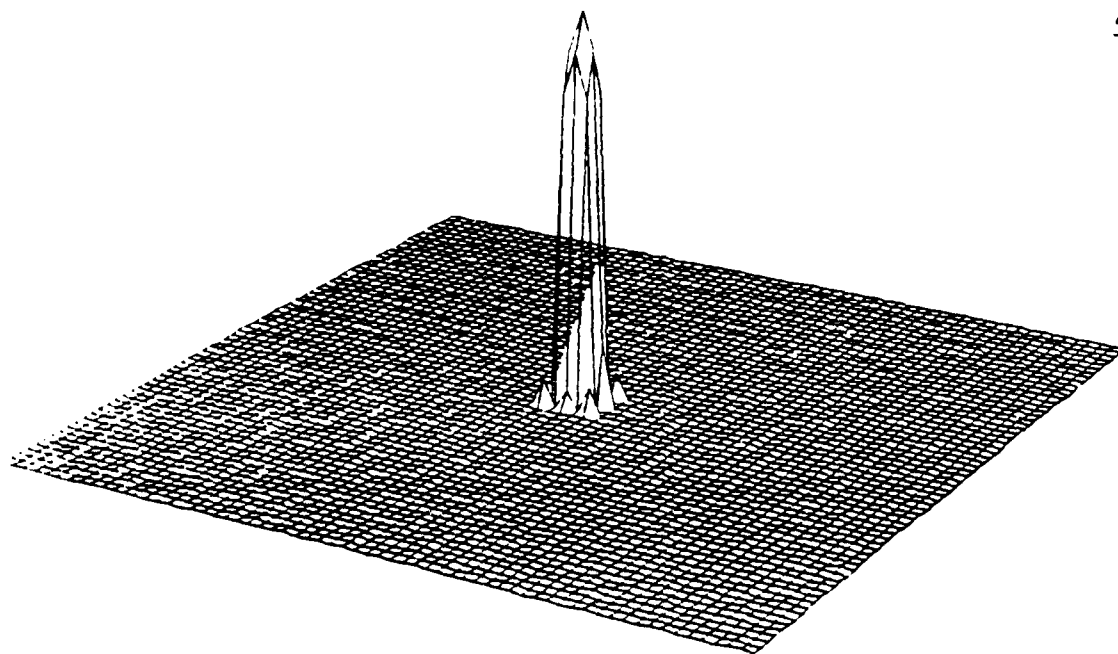


Figure 5.5 a) : Point target at (0,0.).

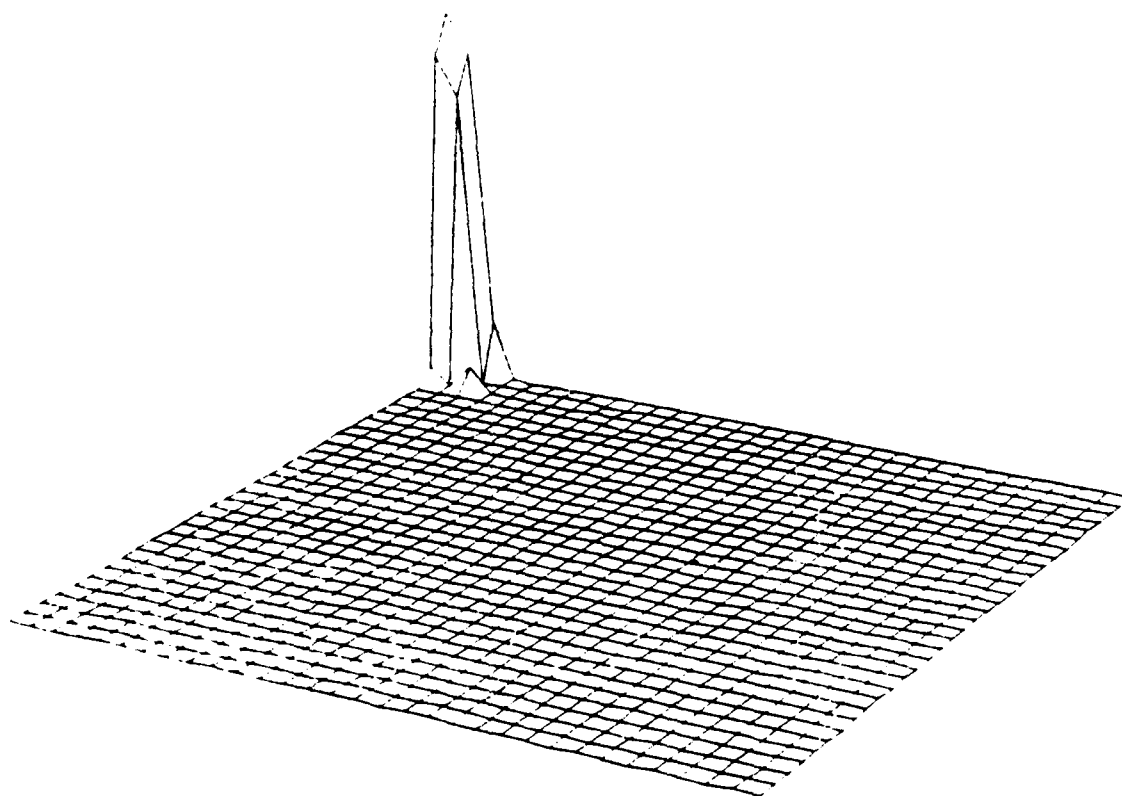


Figure 5.5 b) : Subimage of Figure 5.5 a) at lower right corner.

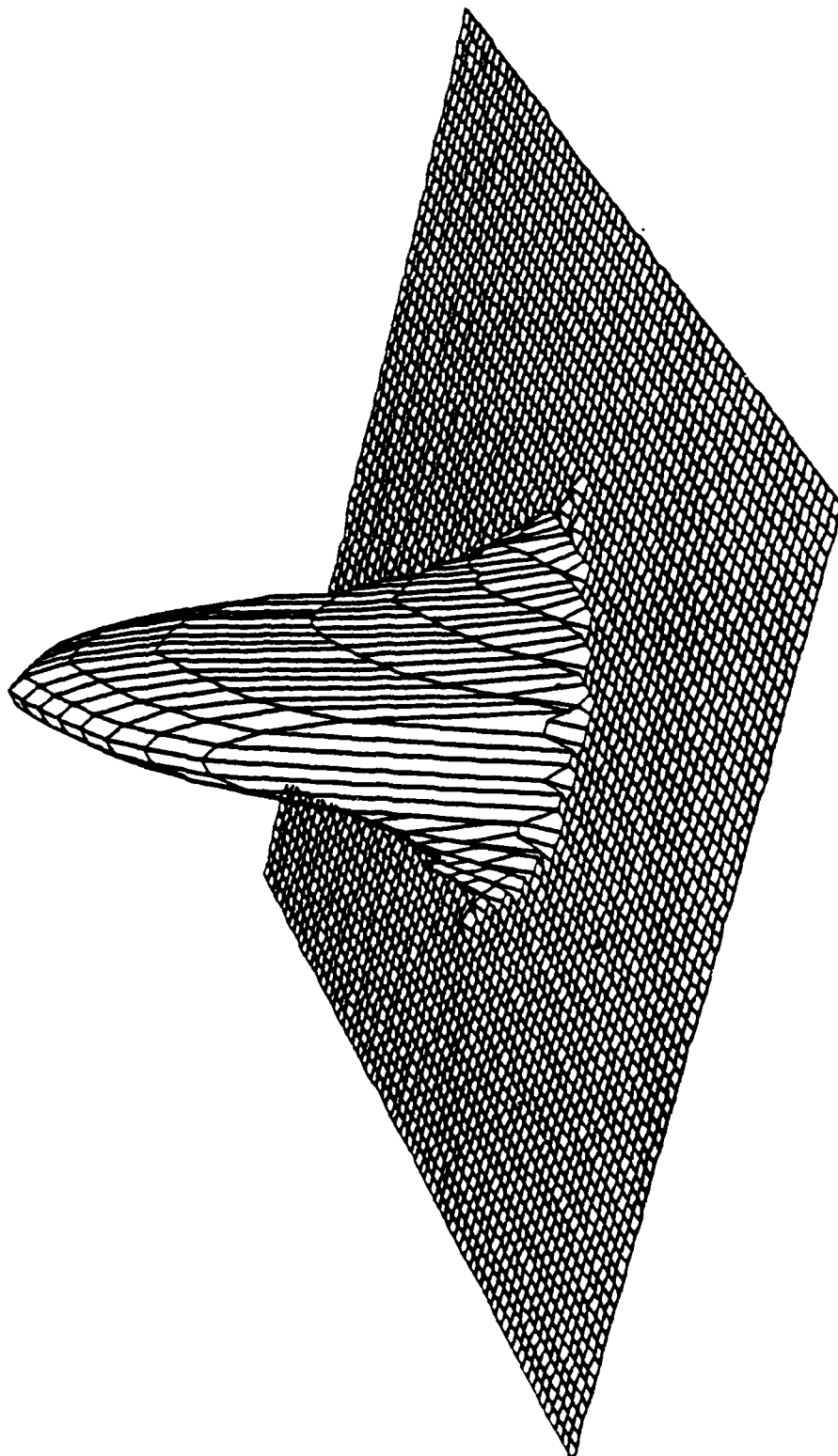


Figure 5.6 : Zoom of Figure 5.4 b).

resulting in substantial savings of computational time. Subregions can be reconstructed by the Fourier based methods by computing the DFT summations instead of using FFT. However, the purpose of using the FFT to reduce the complexity is destroyed. An alternate technique would be to filter and decimate the given Fourier data and then use a lower order FFT corresponding to the size of the subimage desired.

### 5.1.2 Signal Processing Aspects

As discussed in Chapter 3, a 1-D interpolation must be performed prior to back-projection step for each pixel point of the image. The order of the interpolator required can be reduced, if the sampling rate of the transformed projection is increased by zeropadding each projection prior to 1-D inverse transform. If a sequence of length  $N_s$  is zeropadded by  $m N_s$  zeros, and the new sequence of length  $(m + 1) N_s$  is 1-D transformed, then the new transformed sequence will have  $m$  sample points in between every two sample points of the nonpadded transformed sequence [51,11]. A lower order of interpolator can be used on this denser transformed projection. Thus the linear interpolation for  $N^2$  pixels is reduced to a Nearest Neighbor interpolation at the expense of increasing the size of the FFT.

Figure 5.7 a) - d) shows the results of zeropadding the original transformed projection  $\{Q_k\}$  of length 64 with various integer multiples of 64 and using the Nearest Neighbor interpolation scheme. The MNR ratios of these are presented in Table 5.1 along with the MNR ratio for linear interpolation and no zeropadding. The artifacts or the spurious targets present in these figures are due to the changes in the sampling rate and also due to the order of interpolator used. In each case, however, the mainlobe is distinctly resolved and the energy outside the mainlobe decreases with increase in the number of zeros padded. In the worst case, the sidelobes are below -33.5 dB.

The integration for the back-projection step of Equation 3.8 in Chapter 3 that integrates the contribution of all the projections for each pixel point can be performed using any

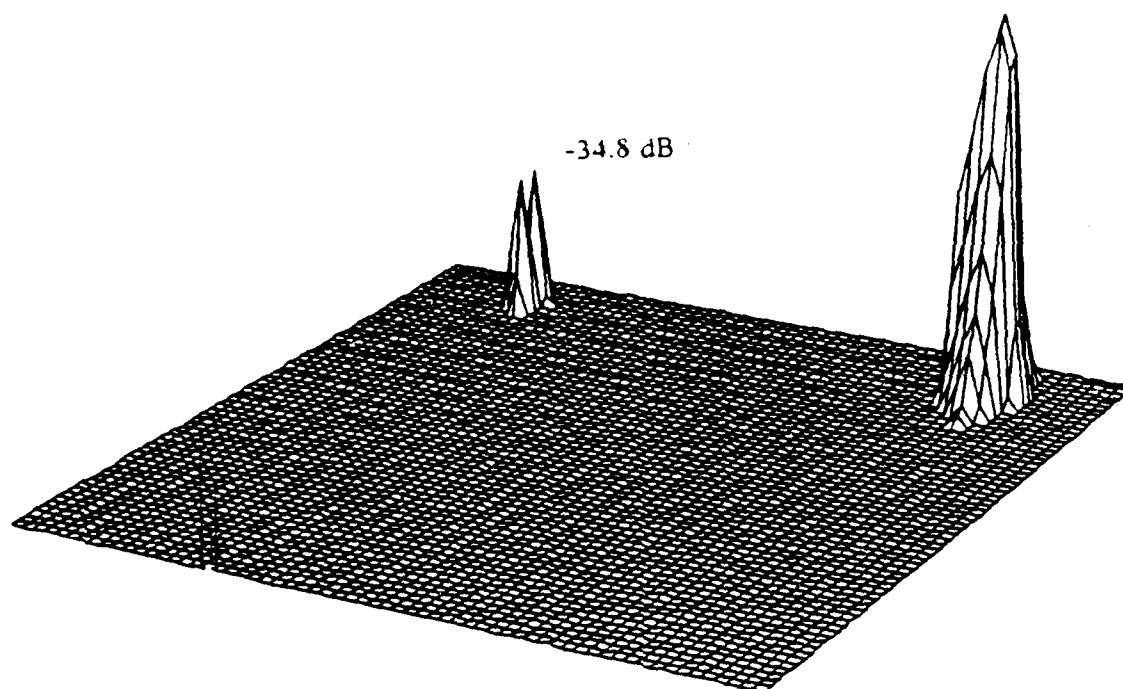


Figure 5.7 a) : Reconstruction using Nearest Neighbor interpolation with no zeropadding.

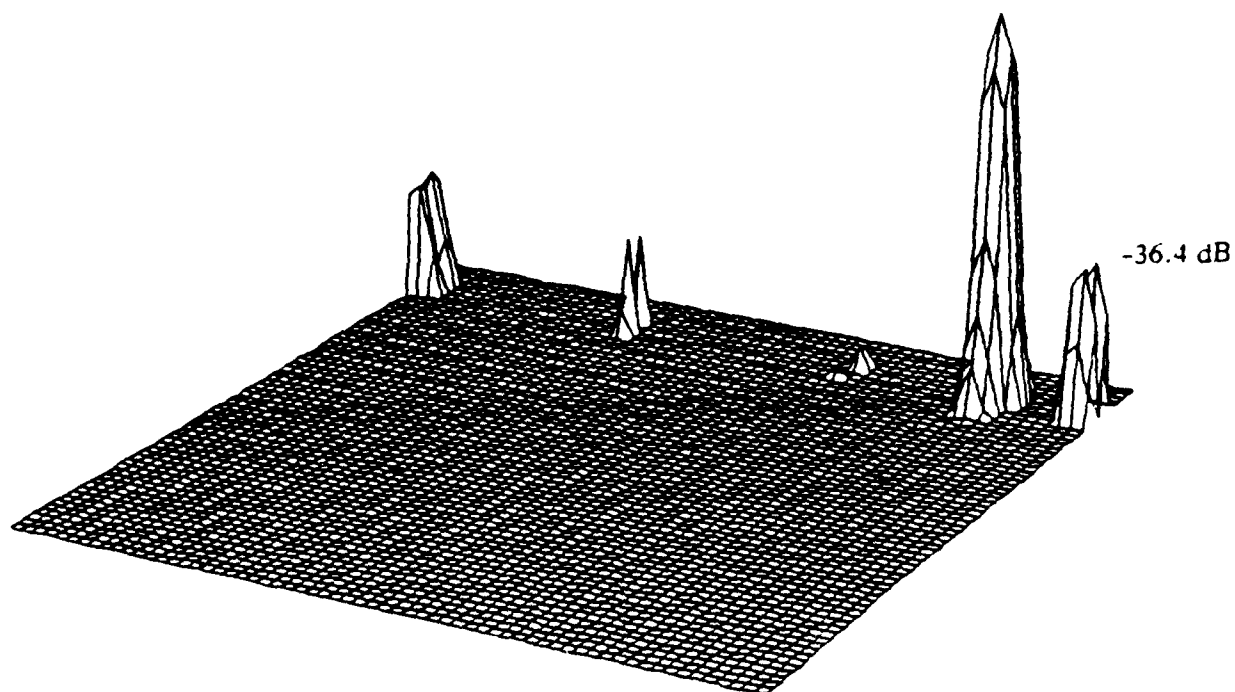


Figure 5.7 b) : Reconstruction using Nearest Neighbor interpolation with 64 zeros padded.

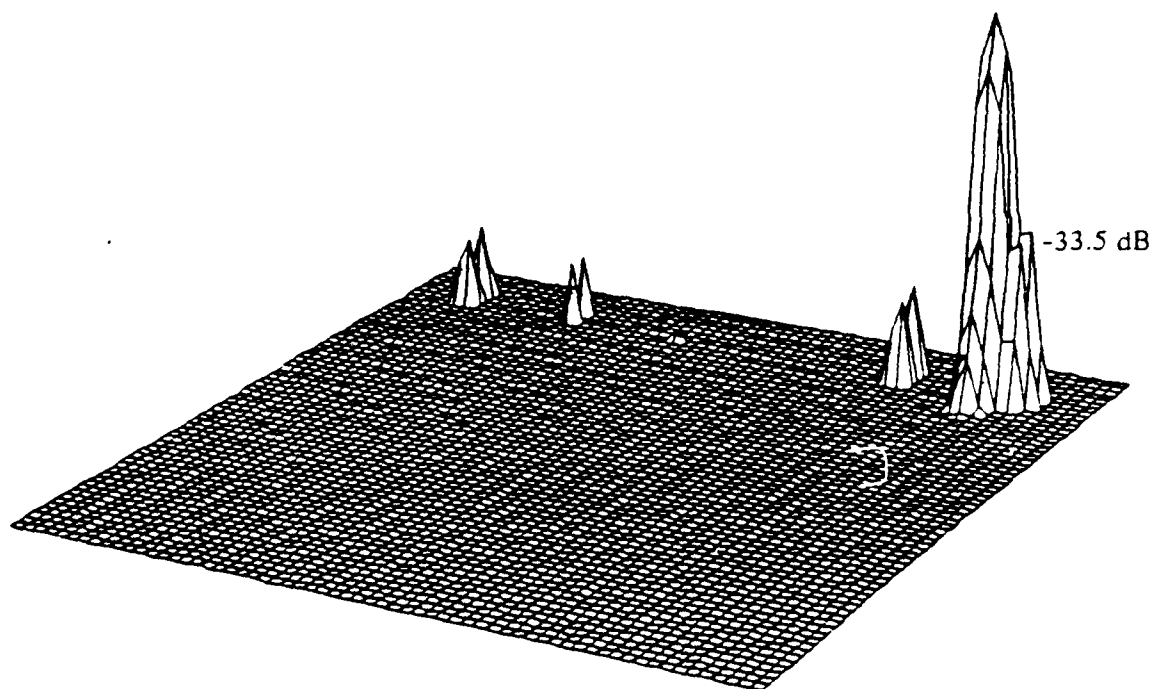


Figure 5.7 c) : Reconstruction using Nearest Neighbor interpolation with 192 zeros padded.

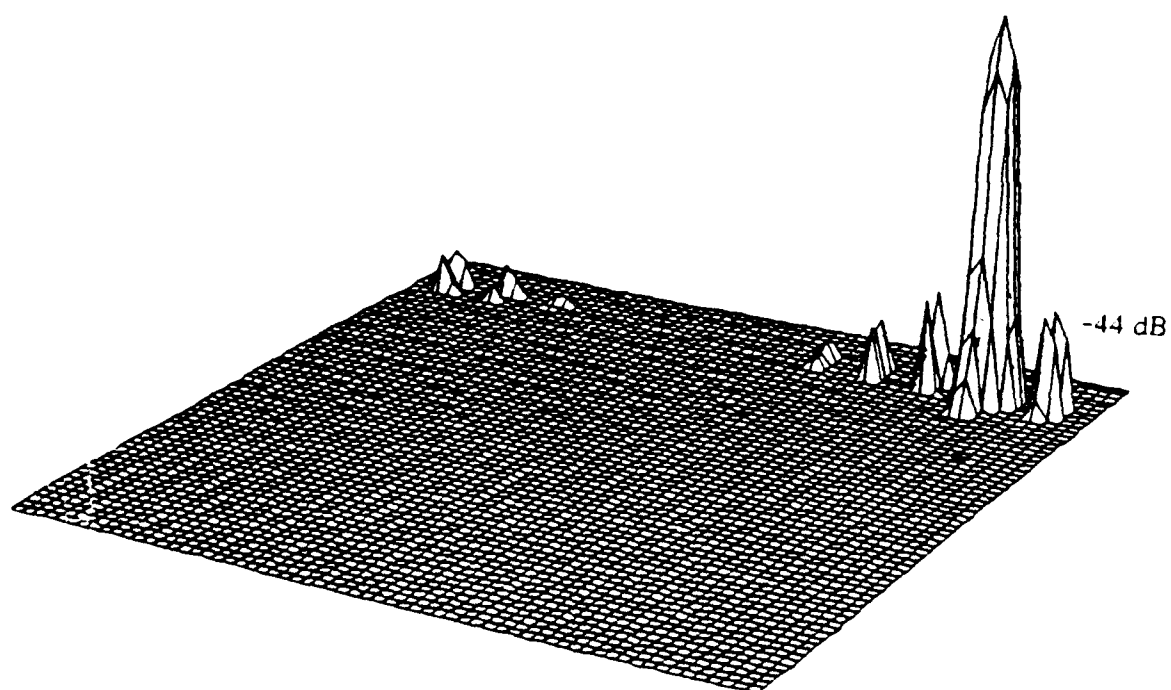


Figure 5.7 d) : Reconstruction using Nearest Neighbor interpolation with 256 zeros padded.

numerical integration technique. The simplest techniques are the rectangular approximation and the trapezoidal approximation which can be illustrated as follows.

Table 5.1 : Summary of MNR values with zeropadding.

Total Length with Zeros Padded	Nearest Ngb. (dB)	Linear (dB)
0	-27.595	-40.38
64	-31.819	...
192	-32.617	...
448	-37.859	...

Consider a function  $f(\theta)$  which has to be integrated over the interval  $[\theta_a, \theta_b]$ . Let  $\{\theta_k : k = 0, 1, 2, \dots, N\}$  be a finite partition of the above interval with  $\theta_0 = \theta_a$  and  $\theta_N = \theta_b$ . Let  $\Delta\theta_k = \theta_k - \theta_{k-1}$  denote the  $k^{th}$  increment. The rectangular approximation can then be expressed as

$$\int_{\theta_a}^{\theta_b} f(\theta) d\theta = \sum_{k=1}^N f(\theta_k) * \Delta\theta_k.$$

while the trapezoidal method can be expressed as

$$\int_{\theta_a}^{\theta_b} f(\theta) d\theta = \sum_{k=1}^N \left[ f(\theta_k) + f(\theta_{k-1}) \right] * 0.5 * \Delta\theta_k$$

The results of using both these approximation techniques indicate there is negligible difference

in using either scheme since the partition of the interval of integration is chosen to be very fine, as shown in Figure 5.8.

Windows are applied to the Fourier domain data prior to the convolution step as discussed in Section 3.3 of Chapter 3. A 2-D separable Hamming window was first applied to the entire data. A 1-D Hamming window was also applied to each projection prior to the 1-D FFT transformation and another along the pixel trajectory during the back-projection step. The results of applying both schemes on various target locations and the two different angles indicate that there is only a slight improvement by using the second scheme. The improvement in MNR was less than .01 %. Both of these schemes seem to have an identical effect on point target reconstructions.

### 5.1.3 Comparisons with FFT Based Method

Assume that an ideal point target is located at  $(x_t, y_t) = (-23, 24)$  in the image plane. To obtain the data for simulation, its Fourier Transform  $G(\omega, \theta)$  defined in Equation 5.2 is sampled on a uniform polar grid as in Figure 5.9. It is assumed that the look angle is  $\theta_m = 3^\circ$  with 1024 equal angular increments and 1024 equal radial intervals over the width  $\Delta\omega = \omega_2 - \omega_1$ , i.e., the data region consists of slightly larger than  $1024 \times 1024$  samples on a polar grid. Each of these projections is centered over the center frequency  $\omega_c$ .

In order to save computation time and resources, the CBP algorithm was applied on subarrays of  $64 \times 64$  samples. This results in  $16 \times 16$  subarrays. Each subarray will be referred by an ordered pair specifying its location on the polar grid. Only two specific subarrays, namely, (2,8) and (16,1) as shown in Figure 5.9 will be considered in this section. Results will, however, vary from subarray to subarray. It is only reasonable to extrapolate the results for  $1K \times 1K$  array from results of many  $64 \times 64$  subarrays.

For the DF<sup>1</sup> case [47], the polar samples were first interpolated to a Cartesian grid using a

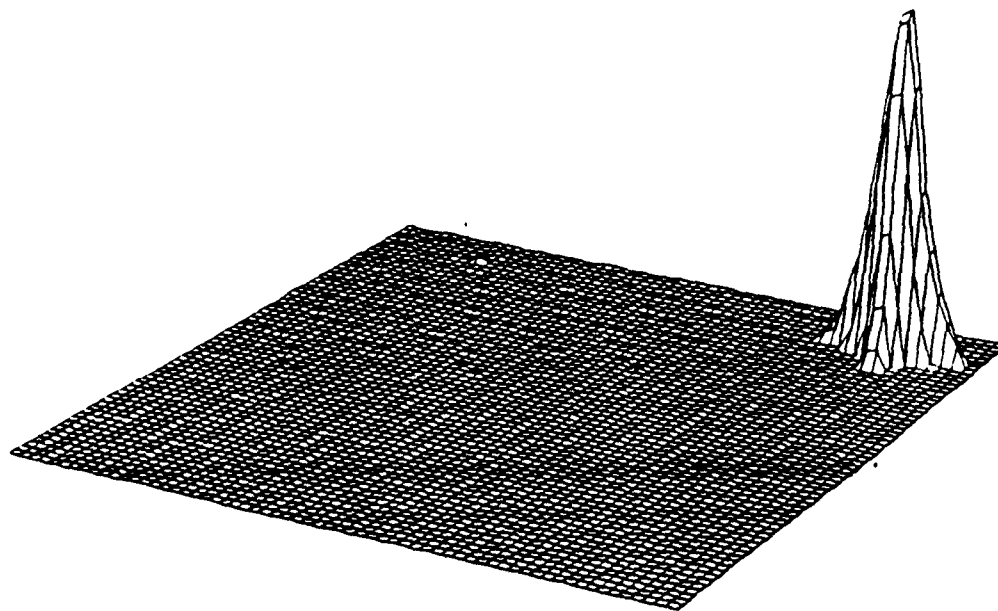


Figure 5.8 a) : Reconstruction using rectangular approximation for the back-projection integral.

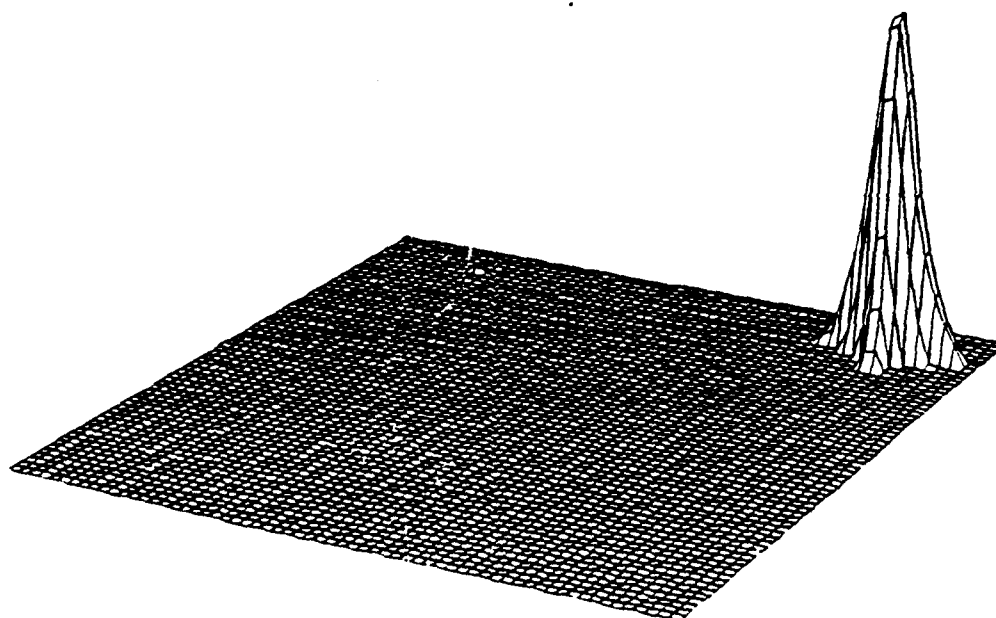


Figure 5.8 b) : Reconstruction using trapezoidal approximation for the back-projection integral.

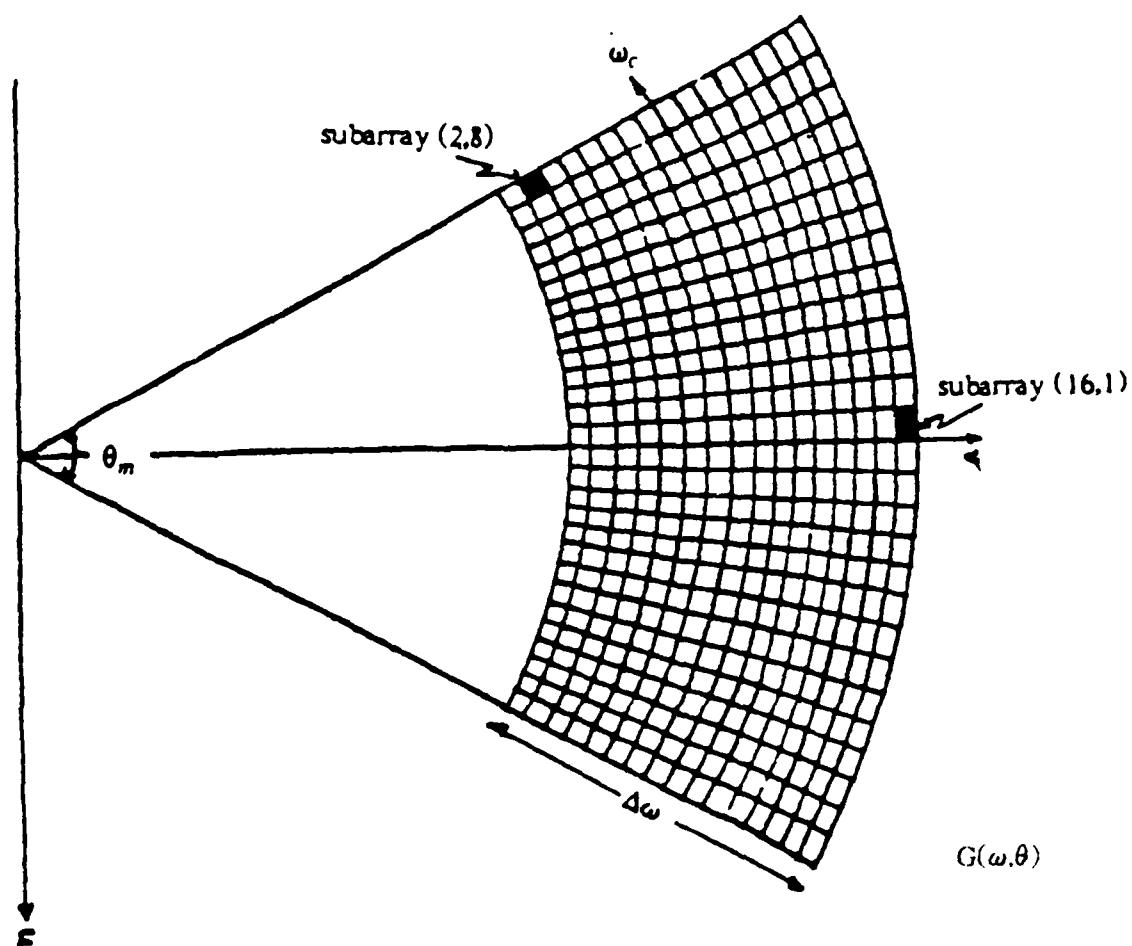


Figure 5.9 : Fourier data indicating location of subarrays.

Standard Polar Format (SPF) interpolator and then processed with a 2-D separable Hamming window prior to applying the 2-D inverse FFT. In this case the Fourier data required in each subarray is slightly larger than the  $64 \times 64$  samples to account for additional data needed for the 2-D polar to rectangular interpolator. The final Cartesian grid in the Fourier region is however  $64 \times 64$ .

Table 5.2 : Comparisons of MNR values for CBP and DF method for subarray (2,8).

Interpolators	CBP (dB)	DF (dB)
Nearest Ngb.	-27.595	-5.64
Linear	-40.384	...
SPF 4 Pt.	...	-24.495
SPF 8 Pt.	...	-36.467
SPF 10 Pt.	...	-42.454

The response for subarray (2,8) is shown in Figures 5.10 and 5.11. Figure 5.10 shows the results of using the Nearest Neighbor interpolation scheme for both methods. With the modified convolution back-projection method, the location of the target can at least be approximately detected. However, that is not true for the Direct Fourier method. Figure 5.11 is the

---

<sup>1</sup>DF results are due to B. Mather and Dr. D. C. Munson, Jr. [17].

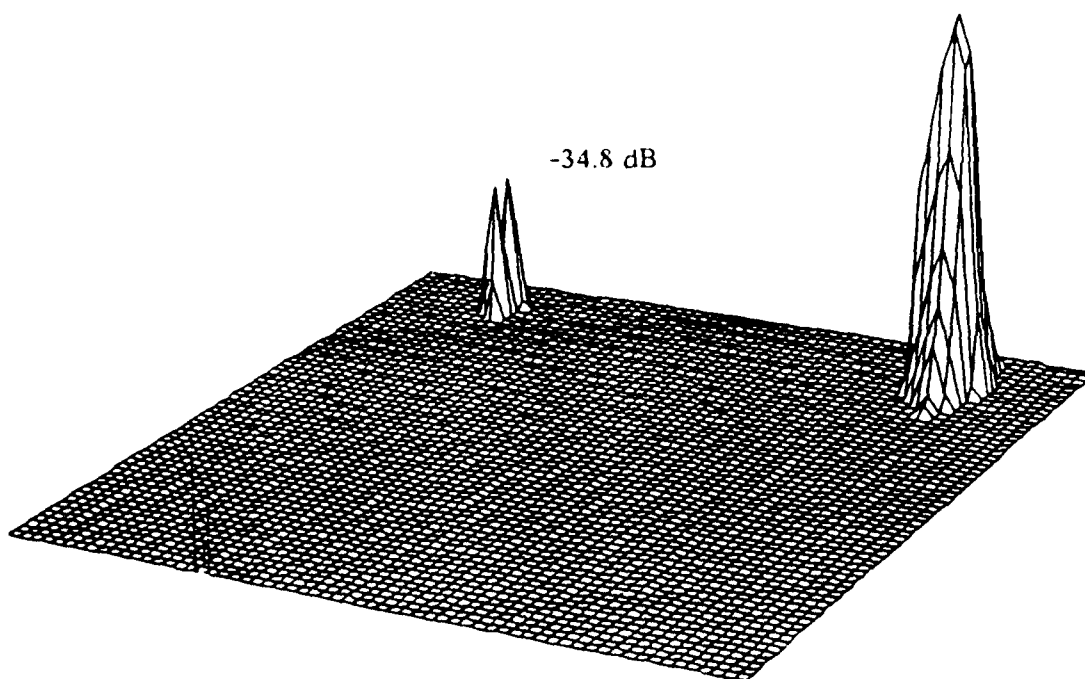


Figure 5.10 a) : Reconstruction using CBP method with Nearest Neighbor interpolation for subarray (2.8).

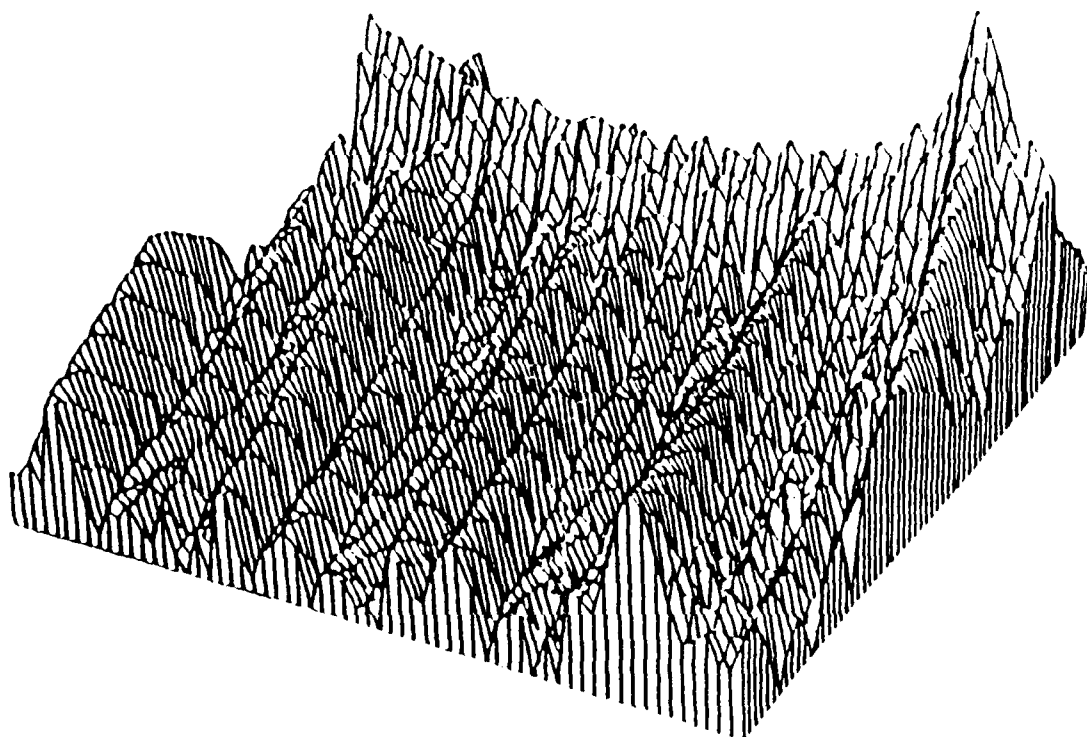


Figure 5.10 b) : Reconstruction using DF method with Nearest Neighbor interpolation for subarray (2.8).

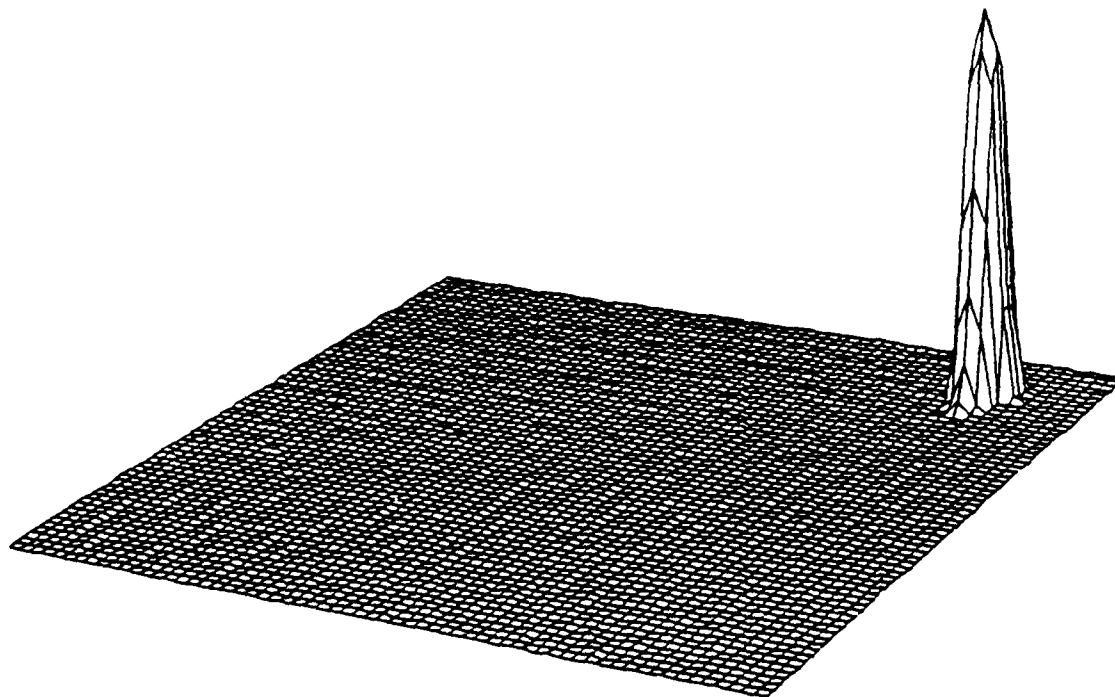


Figure 5.11 a) : Reconstruction using CBP method with linear interpolation for subarray (2.8).

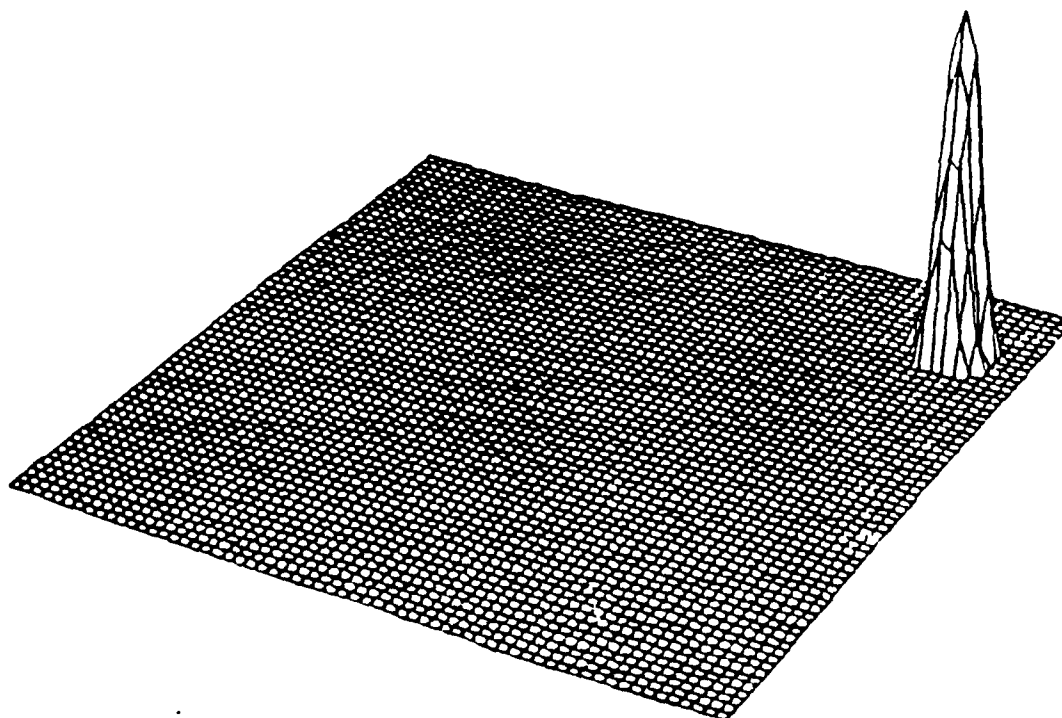


Figure 5.11 b) : Reconstruction using DF method with SPF interpolator of order 9 for subarray (2.8).

comparison of the linear interpolation scheme with modified CBP method and the SPF of order 9 with the DF method. Results of various orders of interpolators for subarray (2.8) are summarized in the Table 5.2. Similarly, Figures 5.12 and 5.13 are the comparisons of the two methods for the subarray (16.1). Note in Figure 5.9 that for subarray (16.1), the interpolators would have to be more accurate than for subarray (2.8). Since the sample spacing in the angular direction is changed when the data are interpolated from polar to Cartesian grid [47], the spurious targets are evident in Figure 5.12 (b) and Figure 5.13 (b). Results for subarray (16.1) are summarized in Table 5.3.

Table 5.3 : Comparisons of MNR values for CBP and DF method for subarray (16.1).

Interpolators	CBP (dB)	DF (dB)
Nearest Ngb.	-21.15	-1.97
Linear	-32.799	...
SPF 6 Pt.	...	-10.62
SPF 10 Pt.	...	-17.01
SPF 14 Pt.	...	-24.48

In conclusion, CBP produces high quality images using a linear interpolation. Equivalent results for the DF case would require interpolators of order between 8 and 10 for subarray

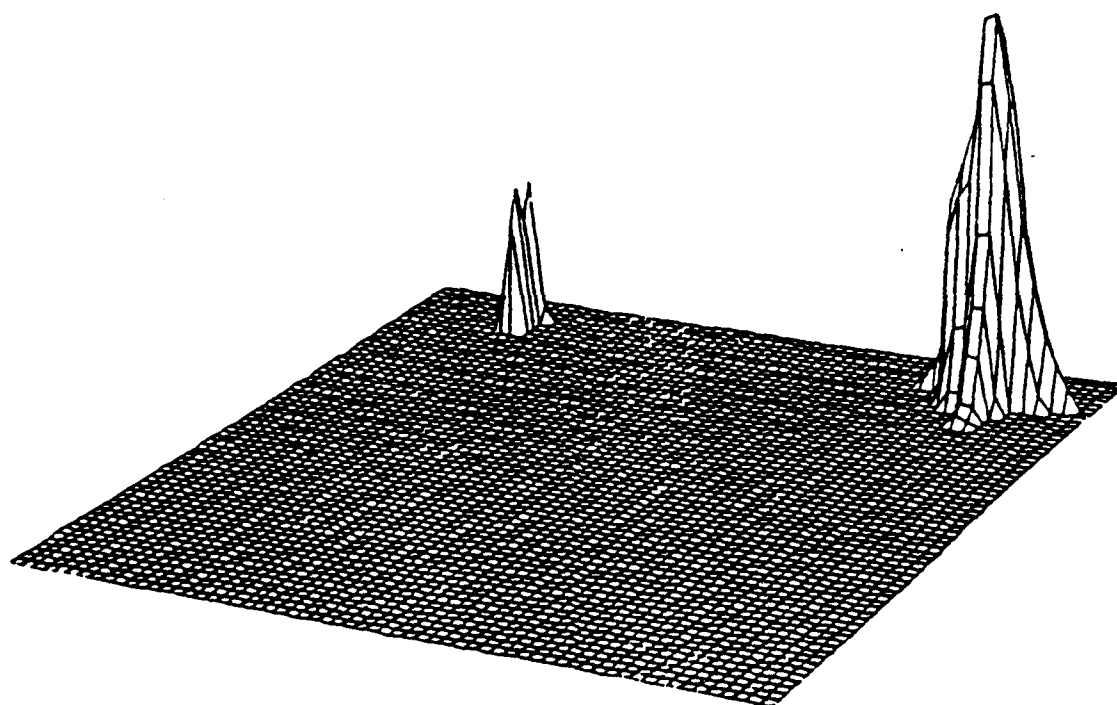


Figure 5.12 a) : Reconstruction using CBP method with Nearest Neighbor interpolation for subarray (16.1).

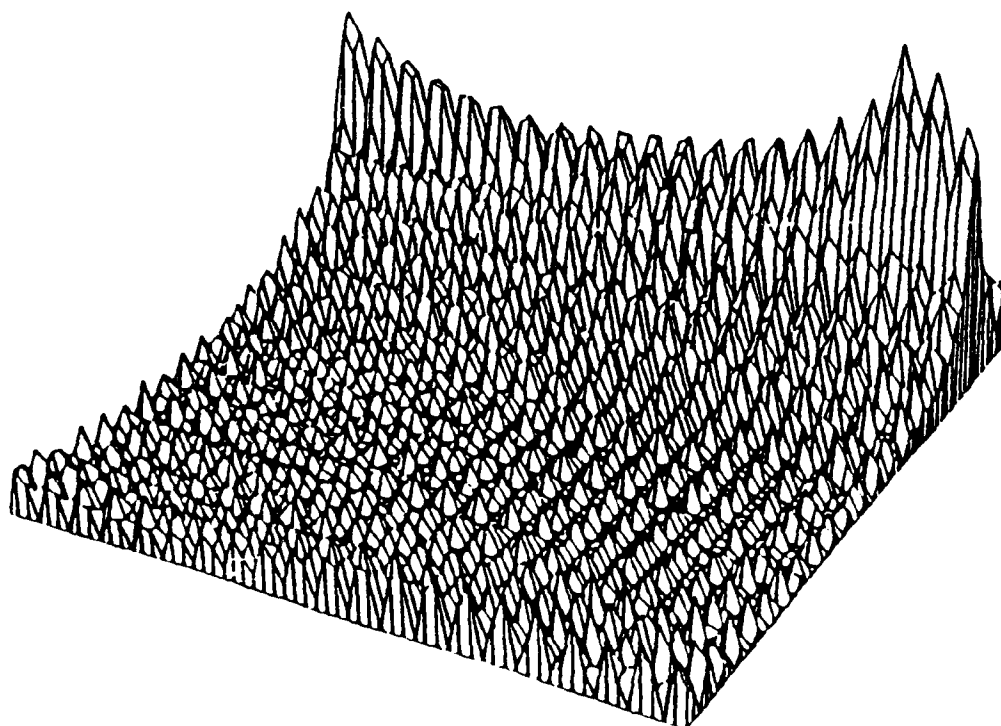


Figure 5.12 b) : Reconstruction using DF method with Nearest Neighbor interpolation for subarray (16.1).

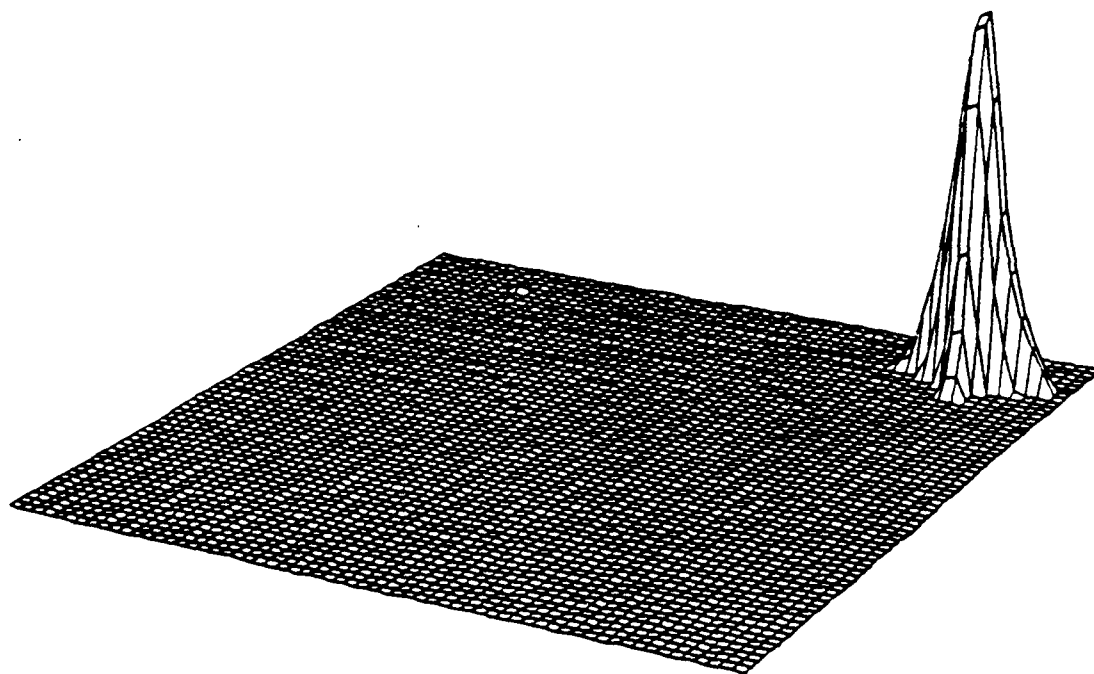


Figure 5.13 a) : Reconstruction using CBP method with Linear interpolation for subarray (16.1).

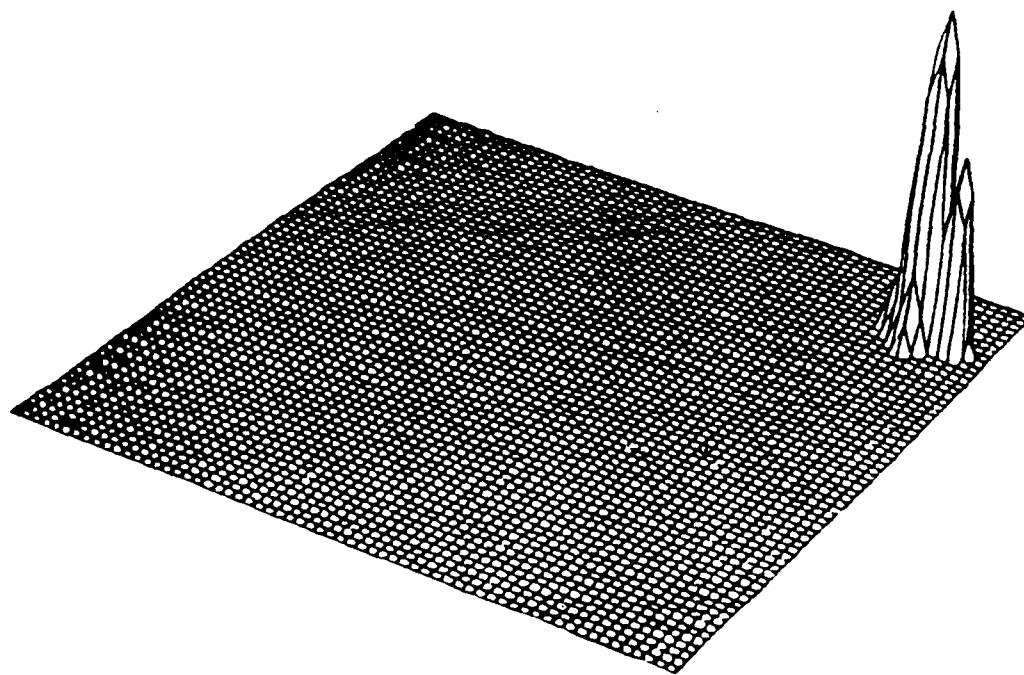


Figure 5.13 b) : Reconstruction using DF method with SPF interpolator of order 14 for subarray (16.1).

(2.8) and of order between 16 and 18 for subarray (16,1). Such high order interpolators would involve a large amount of computation. However, the CBP algorithm will still be more computationally intensive.

The computational complexity of CBP algorithm, as described in Chapter 3, is  $O(N^3)$ , while for the DF method it is  $P(N) + O(N^2 \log_2 N)$ , where  $P(N)$  denotes the complexity of the 2-D interpolator used. The contribution of  $P(N)$  will increase as the order of interpolator increases. Both the CBP algorithm and the DF algorithm were implemented and executed on a VAX-11/780 computer with the UNIX operating system. The CPU times taken by both algorithms are presented in Table 5.4. This demonstrates that the CBP algorithm is around 3 to 5 times slower than the DF algorithm.

Table 5.4 : CPU time comparisons for CBP and DF methods.

Interpolators	CBP (sec)	DF (sec)
Linear	295	...
SPF 9 Pt.	...	58
SPF 17 Pt.	...	100

### 5.1.4 Point Target Response

Consider an ideal point target located at various positions within the target field including the reference point (0,0). The annular region in the Fourier domain with angular width  $\theta_m = 3^\circ$  centered around the  $\theta = 0^\circ$  axis in the  $(\omega, \theta)$ -plane was first considered. The width of the annular region is 19.23 rads/meter. The entire annular region was divided into 64 projections and 64 samples per projections. A separable Hamming window was applied in both angular and radial direction. Note that this is a change of methodology from that of the previous sections. In the earlier sections, the entire annular region was divided into  $1024 \times 1024$  samples and each sub region was composed of  $64 \times 64$  samples. This resulted in  $16 \times 16$  subregions. To evaluate the performance of the CBP algorithm on the entire annular region in the Fourier domain one could use the algorithm directly on the  $1024 \times 1024$  polar grid. This is however computationally expensive and hence the Fourier region is divided into  $64 \times 64$  samples. This amounts of taking every 16th projection and 16th sample per projection from the  $1024 \times 1024$  polar grid. In effect the sampling rate in the Fourier domain is reduced and hence the size of the image region is also reduced. Note that the resolution cell width however is unaffected.

The responses for targets at various locations on the horizontal, the vertical, and on a diagonal are shown in Figures 5.14-5.16. These figures indicate that the widths of the main lobes of the targets increase preferentially along the horizontal direction as opposed to the vertical direction. Furthermore, the widths of the mainlobes were fairly constant with distance of target location from the origin along the vertical direction. This preferential behavior is due to the location of the data in the Fourier domain [46,49]. The MNR values for each of these cases is plotted versus the target coordinates in Figure 5.17. The above experiment is repeated for the Fourier data centered around the  $45^\circ$  axis in the  $(\omega, \theta)$ -plane. The MNR values in this case are plotted in Figure 5.18. From these values one can draw the conclusion that the widths of the mainlobes of the targets are symmetrical in both directions and increase only with the distance

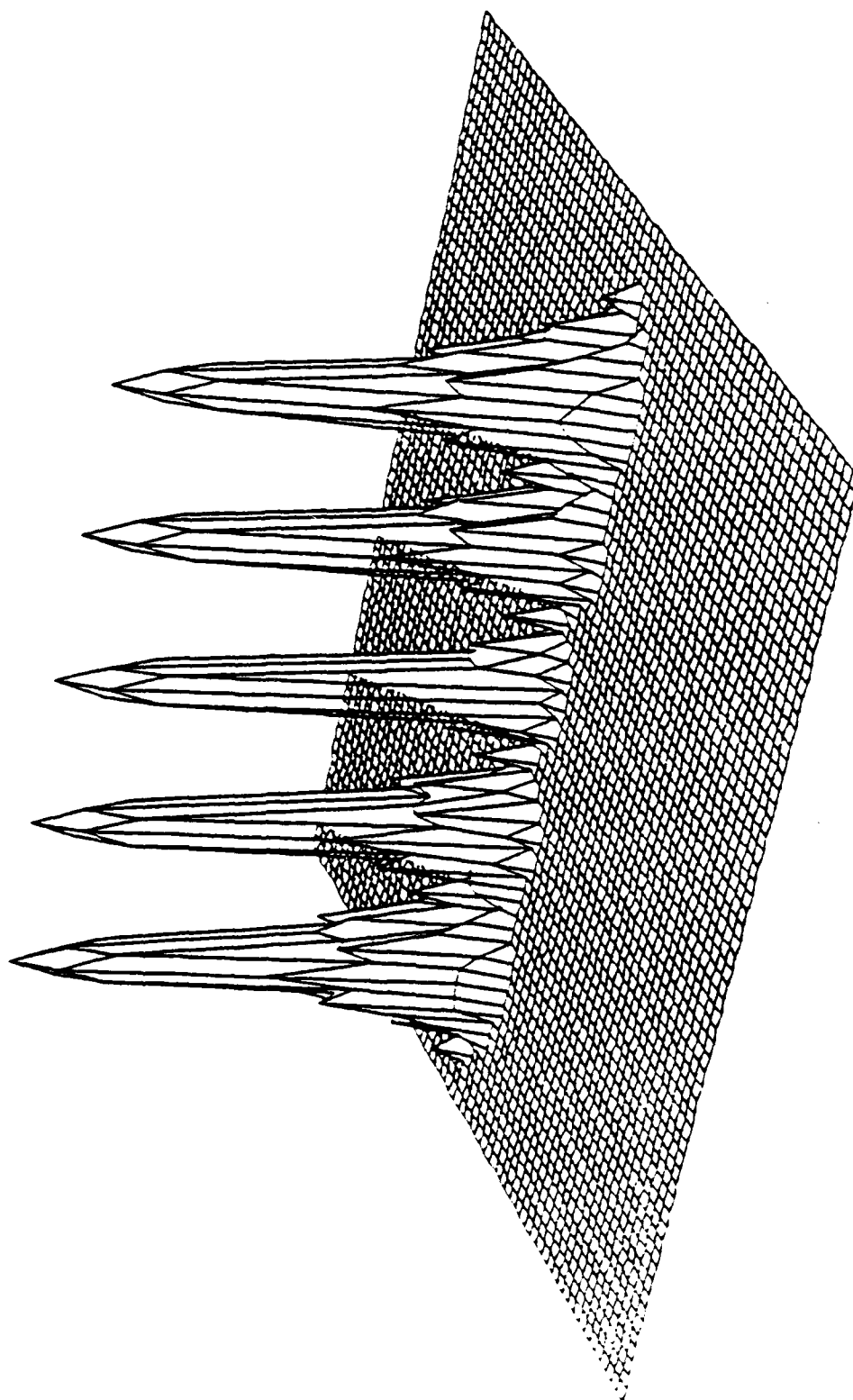


Figure 5.14 : Point targets along horizontal axis for ideal data centered at  $0^\circ$ .

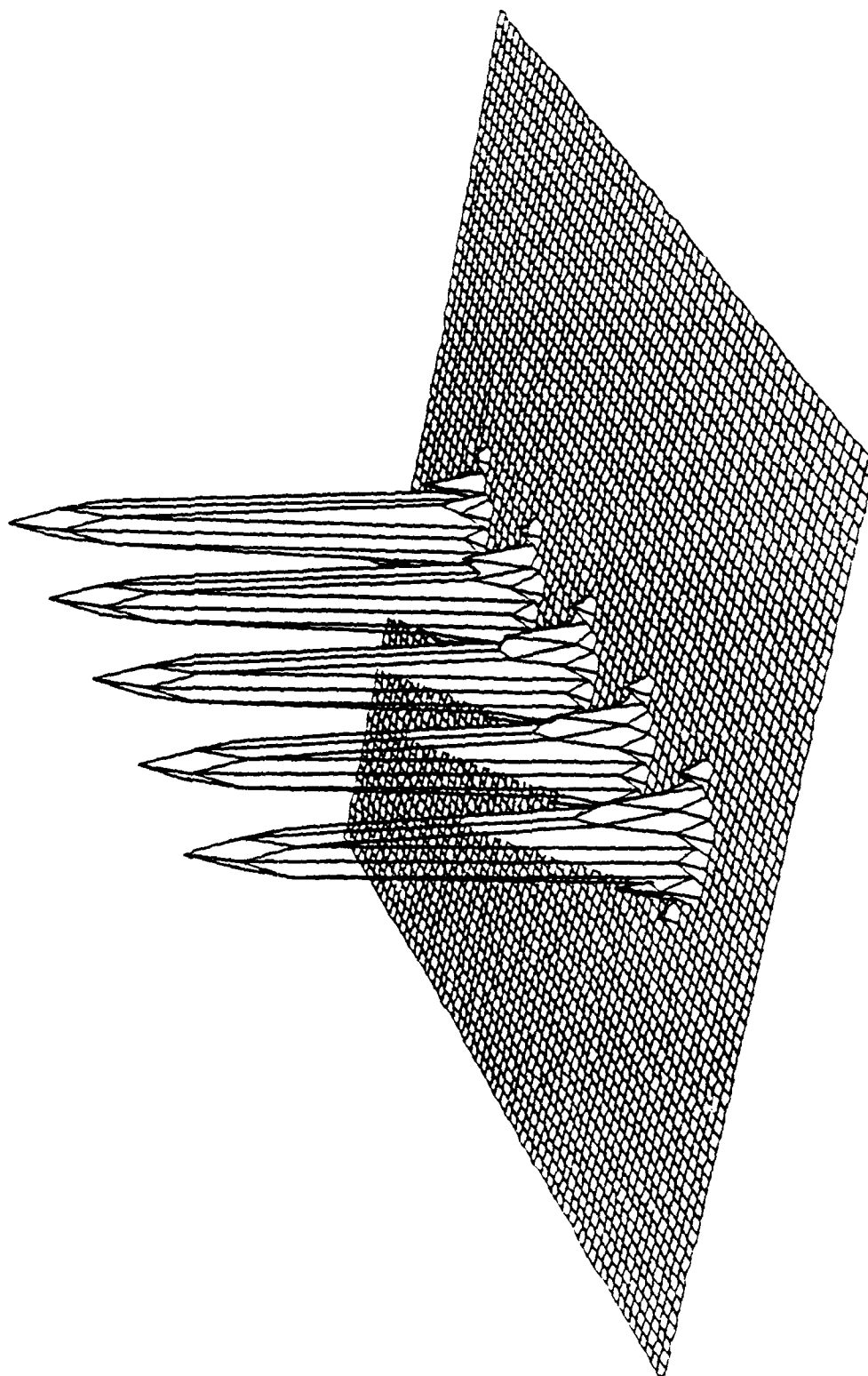


Figure 5.15 : Point targets along vertical axis for ideal data centered at  $0^\circ$

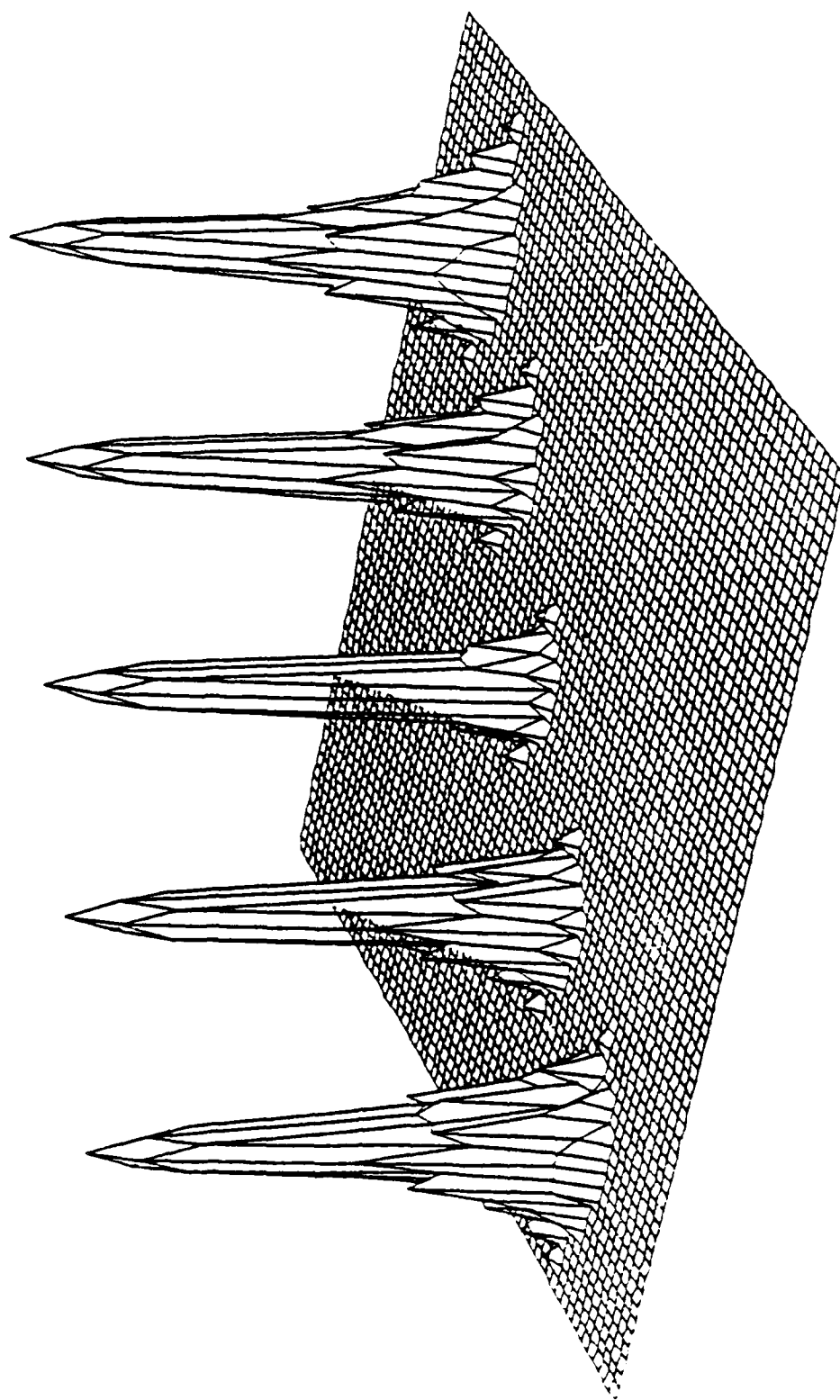


Figure 5.16 : Point targets along diagonal axis for ideal data centered at  $0^\circ$ .

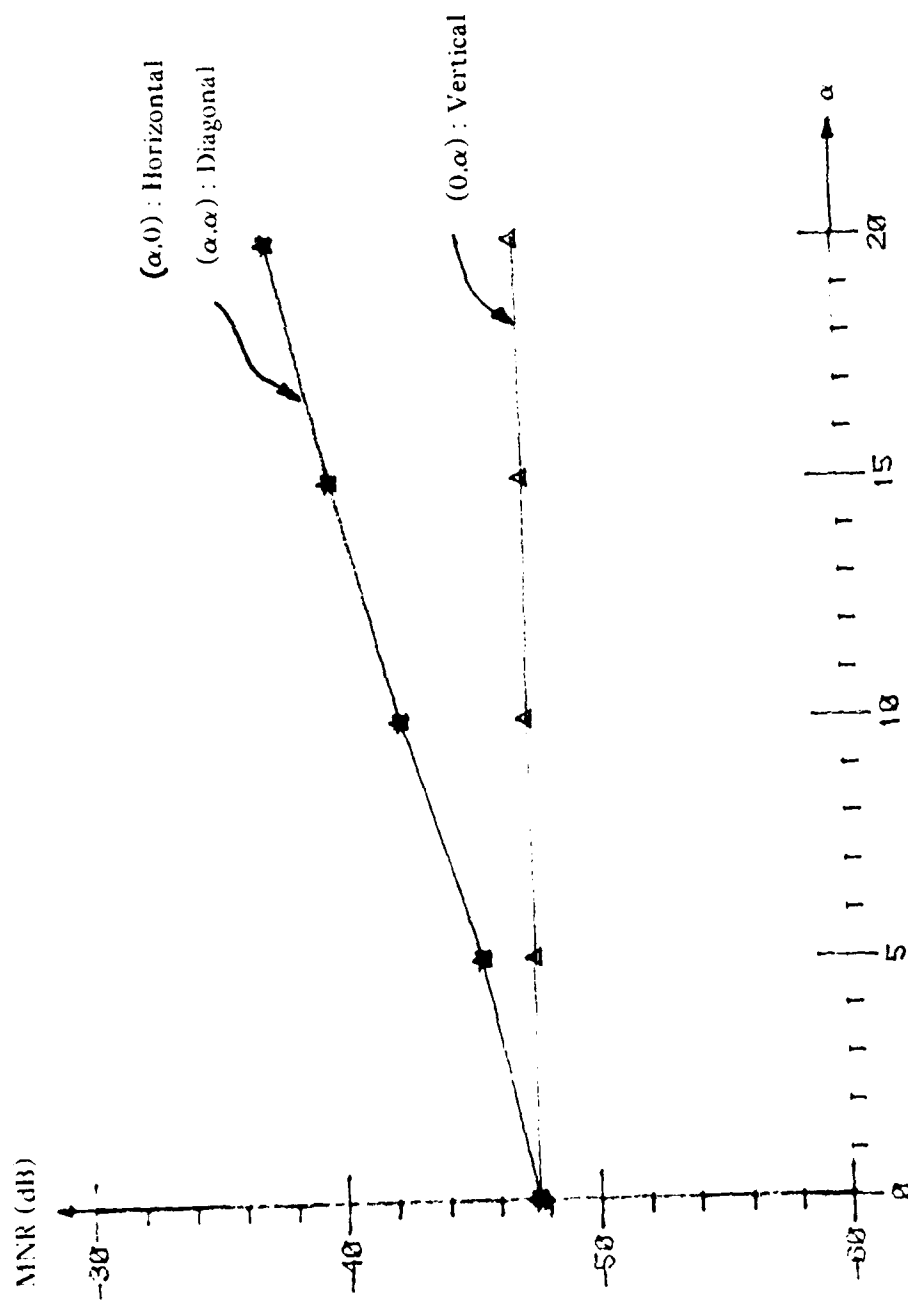


Figure 5.17 : Plot of MINR versus target coordinates for ideal Fourier data centered at  $0^\circ$ .

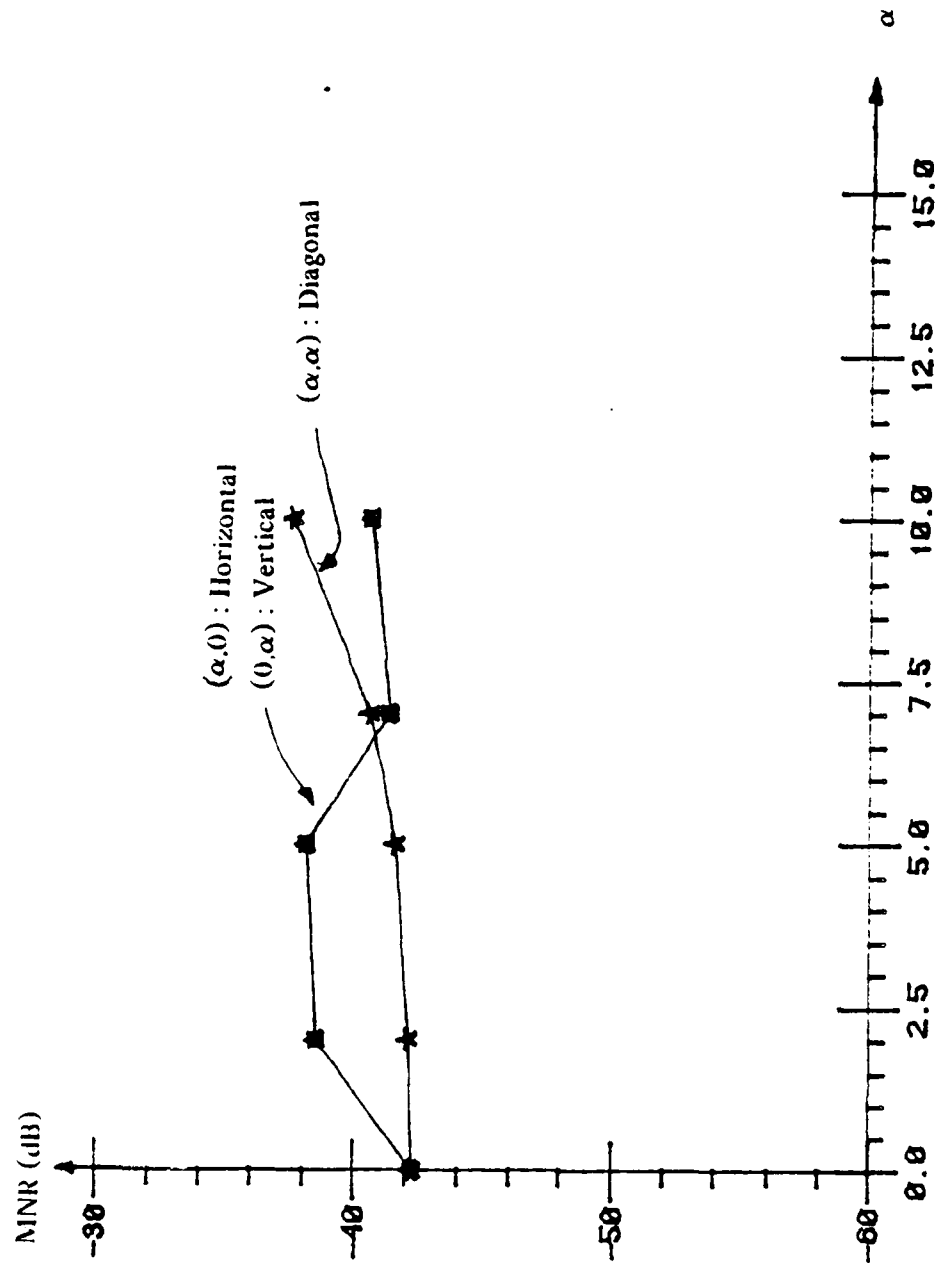


Figure 5.18 : Plot of MINR versus target coordinates for ideal Fourier data centered at  $45^\circ$ .

from the origin.

## 5.2 Non-Ideal Data

In an attempt to make the simulation more realistic, an actual point target response was computed from a mathematical model of the radar return. This mathematical model was developed by Dr. Zeoli at the Hughes Aircraft Co. and the derivations of the model are presented in a Lockheed report by Jenkins and Weis [55]. Only the final result of the model is presented here.

Once again, consider a point target with unit amplitude. As the plane moves along the trajectory, the  $n^{th}$  transmitted pulse of linear FM type at an angle  $\theta_n$  of Equation 2.2.1 will be

$$S_n(t) = \begin{cases} e^{j\omega_0 t + \frac{\gamma}{2}(t-t_n)^2} & |t-t_n| < \frac{T}{2} \\ 0 & \text{otherwise} \end{cases}$$

Then the return signal from the point target is mixed with the reference signal and low-pass filtered to give a unit amplitude signal represented as  $e^{j\Phi(t, \theta_n)}$ , where the phase function  $\Phi$  is derived in [55.52] to be

$$\Phi(t, \theta_n) = \frac{2}{c} (r_n - r_{on}) [\omega_0 + \gamma(t-t_n)] - \frac{2\gamma}{c^2} (r_n^2 - r_{on}^2)$$

In the above equation the variable "t" denotes the passage of time. At any time  $t$ , however, the phase function is recorded on a polar raster at a radius  $\omega$  which is defined to be a linear function of  $t$ . Making the change of variable from  $t$  to  $\omega$  in the above equation and after considerable mathematical manipulation the final form of the phase function, now treated as a function of continuous parameters  $\omega$  and  $\theta$ , is derived to be

$$\Phi(\omega, \theta) = \frac{2\gamma Y_{01}\omega}{cK} \left[ y_1 \sin\theta + x_1 \cos\theta + \frac{\rho^2 \sin\theta}{2Y_{01}} \right]$$

This differs from the ideal point target response of Equation 5.2 by the addition of the third

term which appears due to the approximations made in the mathematical model. Here  $\rho$  is the radial distance of the point target from the reference point in the target field,  $Y_0$  is the minimum distance from the radar ground track to the reference point, and  $K = \frac{Y_0 c}{2}$ . The details of the derivation of the above non-ideal phase function can be found in [55] and will not be presented here. In the remainder of this section, this non-ideal point target response is computed for comparisons of the CBP and the Fourier-based reconstruction methods.

The non-ideal simulation of the point target response was performed for three different look angles, namely,  $\theta_m = 3^\circ$ ,  $8.5^\circ$ , and  $17^\circ$  respectively. The corresponding resolution cell widths were computed to be  $\delta = 2.83m$ ,  $1.0m$ , and  $0.5m$ . The other parameters were chosen to be  $f_0 = 1.0 GHz$ ,  $R = 10km$ , with time bandwidth product of 15000 Hz. The results of the simulations in all three cases, for both the CBP and the DF reconstruction techniques, are shown on plots having a 60dB floor. In these simulations, the height of the radar was assumed to be zero, since it only requires a scale change, and 64 projections were computed, each containing 64 samples.

Figure 5.19 (a) shows the simulation result for the  $\theta_m = 3^\circ$  case using the CBP algorithm with 1-D linear interpolation. The corresponding result for the DF method using a 2-D separable third order FIR interpolator is shown in Figure 5.19 (b). The response of the CBP algorithm to targets placed at different locations has also been studied. In this case, the non-ideal data were assumed to be within an annular region of width  $\theta_m = 3^\circ$  centered around the  $0^\circ$ -axis in the Fourier domain. A plot of the MNR values versus location of the point targets along the horizontal, vertical, and  $45^\circ$  diagonal is shown in Figure 5.20. This plot demonstrates that the MNR values for targets away from the origin are fairly constant with distance from the origin in all three cases.

Figures 5.21 (a) and 5.21 (b) show the simulation results for both the CBP and the DF algorithms with the look angle  $\theta_m = 8.5^\circ$ . The corresponding results for the  $\theta_m = 17^\circ$  case are

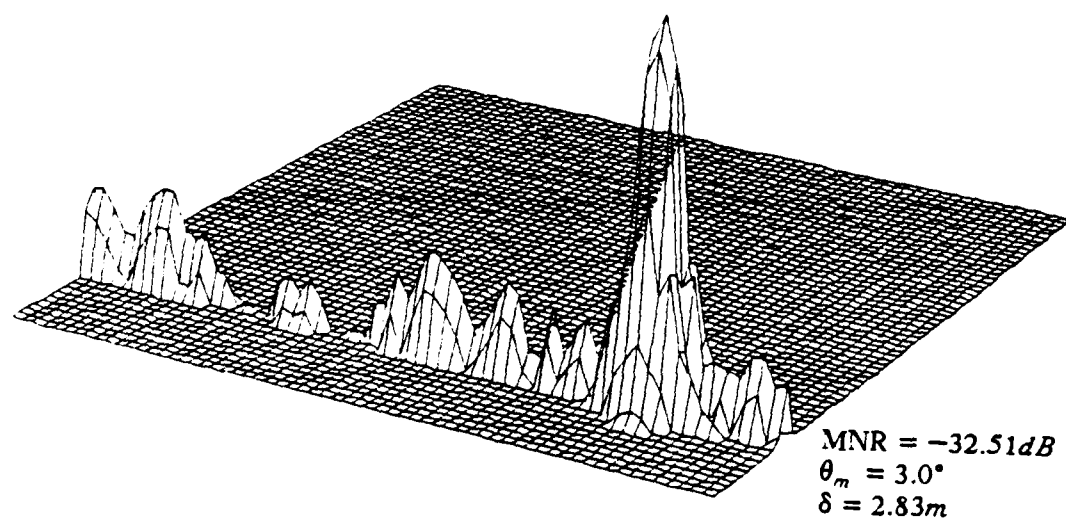


Figure 5.19 a) : CBP reconstruction with non-ideal data.

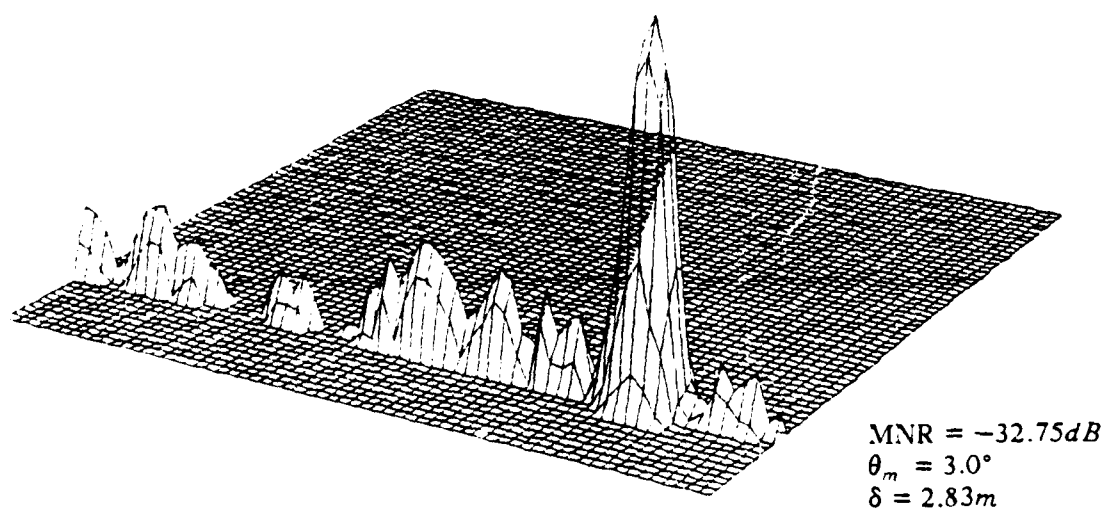


Figure 5.19 b) : DF reconstruction with non-ideal data.

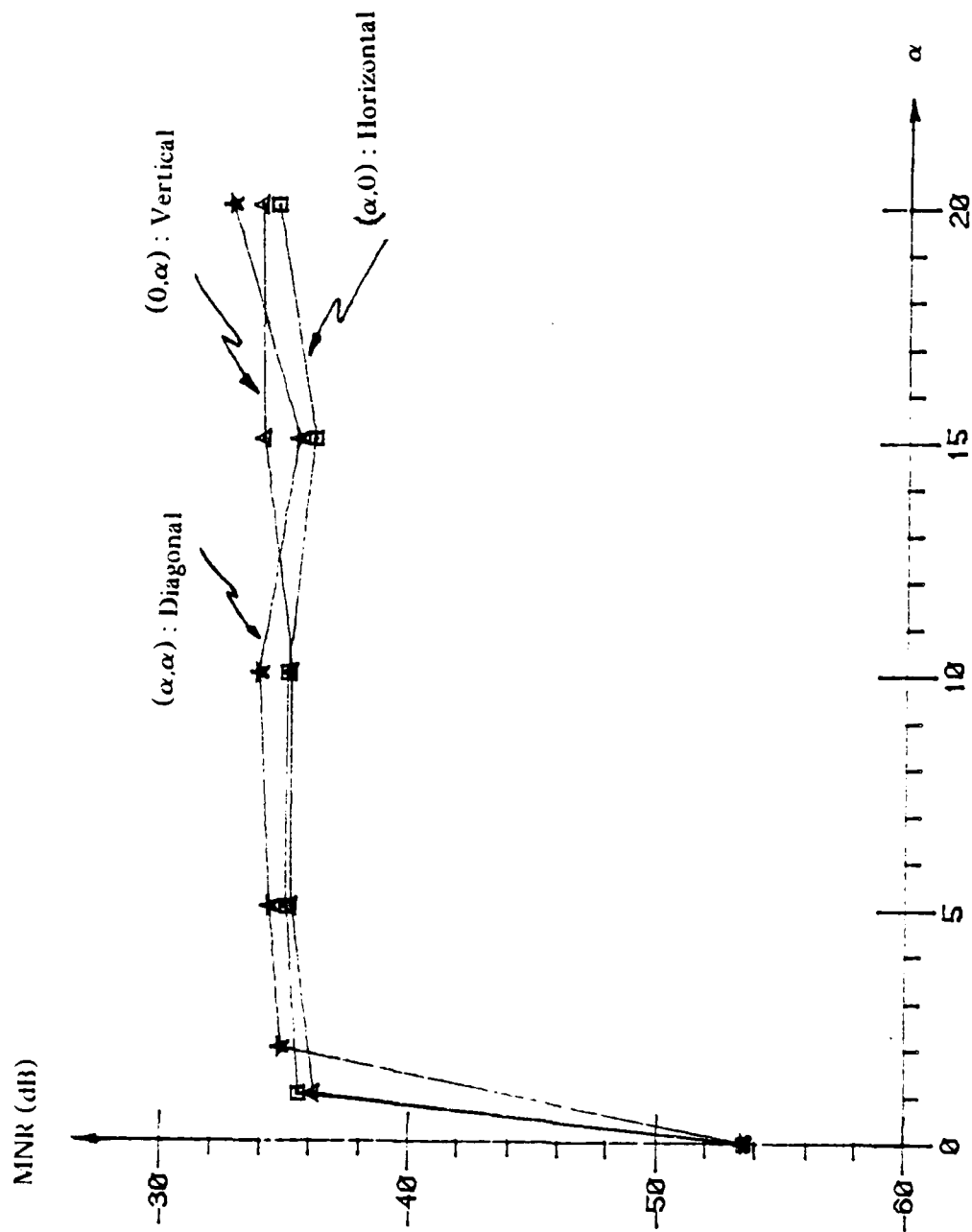


Figure 5.20 : Plot of MINR versus target coordinates for non-ideal Fourier data centered at  $0^\circ$ .

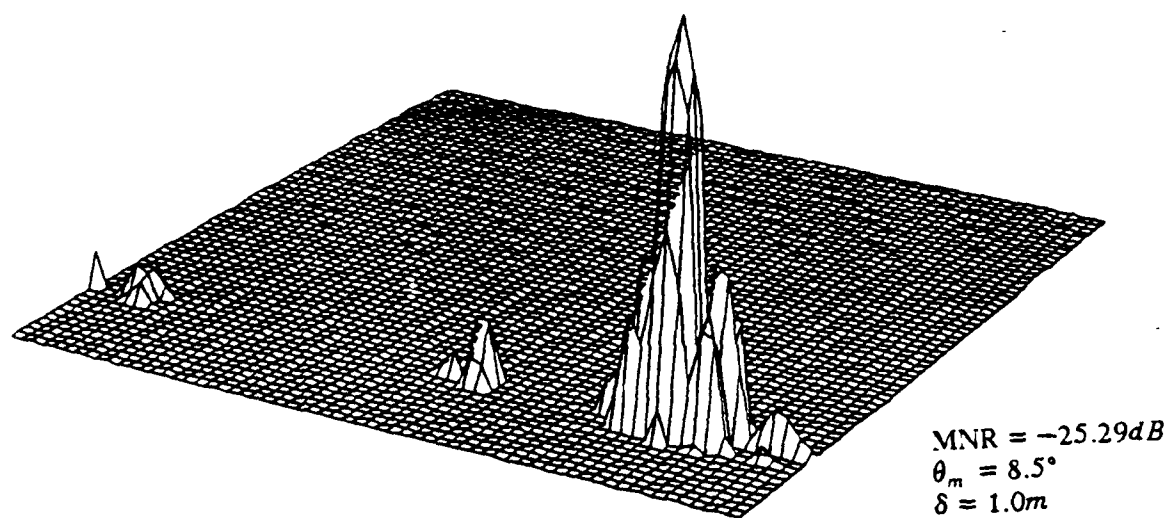


Figure 5.21 a) : CBP reconstruction with non-ideal data.

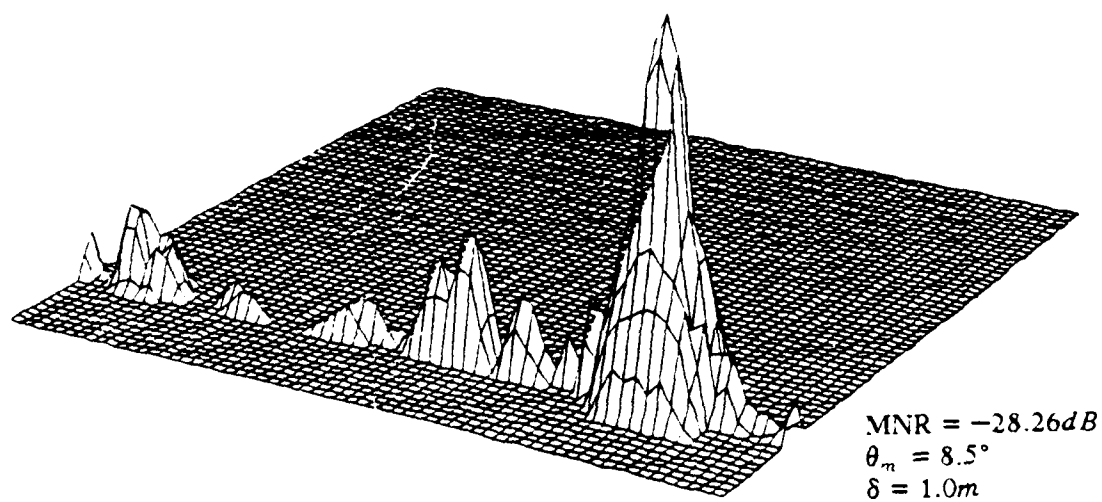


Figure 5.21 b) : DF reconstruction with non-ideal data.

shown in Figures 5.22 (a) and 5.22 (b), respectively. These results indicate that while the two algorithms are quite similar in performance for small look angles, the quality of the image produced by the CBP algorithm improves with wider look angles. The DF algorithm, however, deteriorates in performance as the look angle is increased. This is due to the fact that the difference between the annular region and the rectangular region inscribed within it becomes larger as the look angles increase, and hence this increases the errors caused by 2-D interpolation. The MNR values produced by both algorithms fall in magnitude as the look angle increases. For small look angles the MNR produced by the DF method is higher in magnitude than that of the CBP method. The opposite is true for large look angles.

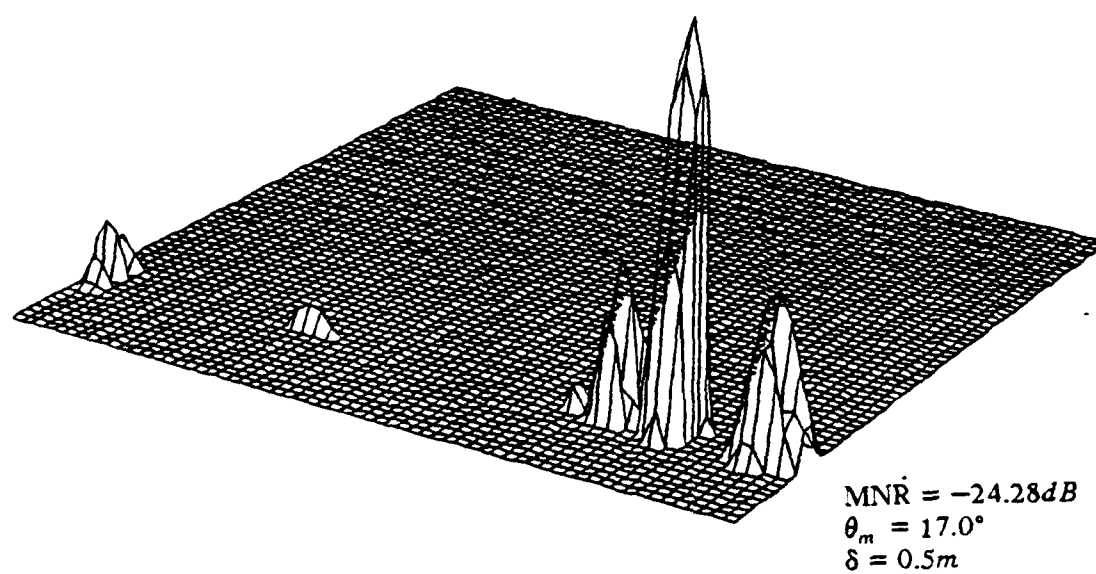


Figure 5.22 a) : CBP reconstruction with non-ideal data.

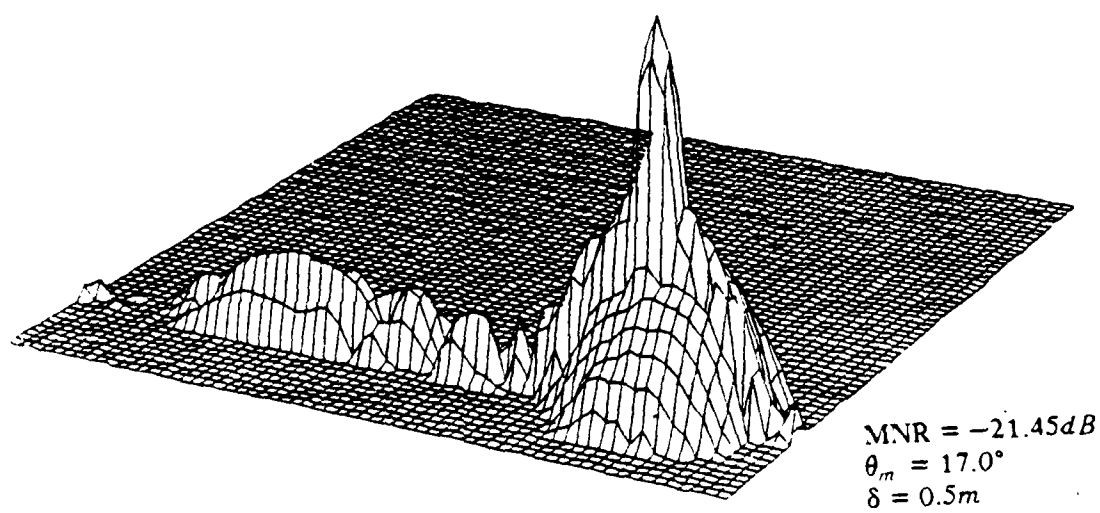


Figure 5.22 b) : DF reconstruction with non-ideal data.

## CHAPTER 6

### CONCLUSIONS

Synthetic Aperture Radar image reconstruction is a problem of producing an image from limited angle samples of the Fourier data region. The proposed CBP algorithm proved to be very easily adaptable to spotlight mode SAR data. A detailed study of the algorithm revealed some useful characteristics, like the ability to produce subimages, image zooming and the ability to reconstruct an image simultaneously with the data collection. These features have significant military and civilian applications.

In addition, a comparison of MNR of CBP and direct Fourier methods on the subarrays of the Fourier data revealed that CBP can produce high quality images without spurious targets. The DF methods, however, use sufficiently high-order 2-D interpolators to produce data on a rectangular grid, and this results in images with spurious targets [29,50]. Modified CBP with 1-D linear interpolation produced an MNR equivalent to the DF method with the order of interpolator being 9 or 14, depending on the subarray chosen.

The CBP algorithm is a point-by-point reconstruction algorithm whose computational complexity is shown to be  $O(N^3)$ , where  $N \times N$  is the array size of the final image. In contrast, the FFT-based algorithms have a complexity of  $P(N) + O(N^2 \log N)$  and the computation time is usually dominated by the 2-D interpolation which has a computational complexity of  $P(N)$ . When  $N$  is large and the interpolator order is small, it is apparent that the CBP algorithm requires more arithmetic operations than the FFT-based methods. However, the inherent parallelism in the CBP algorithm was exploited by using special purpose architectures with  $N$  processors to make computational complexity equivalent to  $O(N^2)$ .

Another important issue in modern SAR is to find high quality reconstruction algorithms that are also well-suited for VLSI circuit realization. Rather than searching for algorithms that require a minimum number of multiplications and additions, it may be more important in VLSI

implementations to find highly parallel algorithms that can be partitioned properly. In general, performance depends largely on the net throughput rate of the data. Therefore, the best SAR algorithms of the future may not necessarily be those with minimal arithmetic operations, but rather will be those which can be partitioned well for VLSI implementation and pipelined properly for high throughput rates [8,9,10].

In this dissertation, the CBP algorithm has been studied from several qualitative and implementational view points, as mentioned above. It has also been demonstrated that this algorithm is well suited for SAR image reconstruction and has several desirable features. However, there are several issues that need to be investigated both to improve the performance of the CBP algorithm in its present form and also to make it readily applicable in physical SAR systems. Some of these issues are discussed below.

First, a detailed study of motion compensation and auto-focusing needs to be performed so that the final image produced is focussed and properly resolved. Since an aircraft cannot fly a perfect straight line trajectory, motion compensation is essential. In addition, a well-focused image requires that the phase of each return be known with high precision. Usually, two subarrays of Fourier data are used to produce two images centered on the same reference point. Due to the phase errors in the measurements, these two images will be displaced with respect to each other. Cross correlation of the two images can give some indication of the phase errors [4]. This problem of auto-focusing may be easily adaptable to the CBP algorithm. An image from, say, ten projections can be cross correlated with the image from the next ten projections, for example, to determine the phase error estimate. The details of this approach need to be studied in the future.

Another topic to be investigated is the process of pre-summing. As discussed in Chapter 5 of this thesis, the Fourier data produced by the SAR system is over-sampled, and then by the process of pre-summing the sampling rate is decreased so that the cost of data storage and the

computation time of the processing is reduced at the expense of the computation involved in pre-summing. In this thesis the CBP algorithm was applied only to the Fourier data obtained after the pre-summing operation. The option of applying the algorithm directly on the raw data without pre-summing needs to be examined. The disadvantage of using the raw data is that the memory requirements for each processor in the architecture described in Chapter 4 will increase by a factor of the over-sampling, and that both the convolution and the back-projection steps have to be performed for a larger number of projections. The effect of the image quality produced by using the CBP algorithm on either the over-sampled raw data or the pre-summed data needs further investigation. The effect of the presence of noise in the SAR system itself and data transmission channels on the quality of the image reconstructed and the robustness of the CBP algorithm to several parametric variations are also some topics for future research.

## REFERENCES

- [1] Adams M. F., and Anderson A. P., "Synthetic aperture tomographic (SAT) imaging for microwave diagnostics," *IEEE Proc.*, vol. 129, pp. 83-88, April 1982.
- [2] Alpers W. R., Ross D. B., and Rufenach D. L., "On the detectability of ocean surface waves by real and synthetic aperture radars," *IEEE Trans. Antennas Propagat.*, vol. AP-27, pp. 635-690, 1979.
- [3] Ausherman D. A., Kozma A., Walker J. L., Jones H. M., Poggio E.C., "Developments in radar imaging," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-20, no. 4, pp. 363-399, July 1984.
- [4] Bendor G. A. and Gedra T. W., "Single-pass fine-resolution SAR autofocus," *IEEE NAECON*, vol. 1, May 1983, pp. 482-488.
- [5] Brown W. M., "Walker model for radar sensing of rigid target fields," *IEEE Trans. Aerosp. Electronic Syst.*, vol. AES-16, Jan. 1980, pp.104-107.
- [6] Brown W. M., "Synthetic aperture radar," *IEEE Trans. Aerosp. Electronic Syst.*, vol. AES-3, no.2, 1967, pp.217-229.
- [7] Brown W. M. and Porcello L. J., "An introduction to synthetic-aperture radar," *IEEE Spectrum*, September 1968, pp.52-62.
- [8] Brookner E., "Trends in radar signal processing," *Microwave Journal*, pp. 1-12, October 1982.
- [9] Brookner E., "Developments in digital radar processing," *Trends & Perspectives in Signal Processing*, vol. 2, Number I, pp. 7-23, January 1982.
- [10] Brookner E., "Synthetic aperture radar processing," *Trends & Perspectives in Signal Processing*, vol. 2, Number I, pp. 7-23, January 1982.
- [11] Crochiere R. E., Rabiner L. R., "Optimum FIR digital filter implementations for decimation, interpolation, and narrow-band filtering," *IEEE Trans. ASSP.*, vol. ASSP-23, no. 5, 1975, pp.444-456.
- [12] Cumming I. G. and Bennett J. R., "Digital processing of SEASAT data," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Proc.*, pp. 710-717, April 1979.
- [13] Chung-Chien Chen and Andrews H. C., *Multifrequency Imaging of Radar Turntable Data*, *Proc. IEEE Tran. on Aerospace and Electronic Systems*, vol. AES-16, no.1, January 1980, pp. 15-22.

- [14] Cutrona L. J., "A high resolution radar combat surveillance system," *IRE Trans. Military Electronics*, vol. MLS-5, pp. 127-131, April 1961.
- [15] Davidson, E. S., "The design and control of pipelined function generators," *Proc. Int. IEEE Conference on Systems, Networks, and Computers*, Oaxtepec, Mexico, January, 1971, pp. 19-21.
- [16] Davison, M. E., "The ill-conditioned nature of the limited angle tomography problem," *SIAM J. Appl. Math.*, vol. 43, no. 2, April 1983, pp. 428-448.
- [17] Desai M. and Jenkins W. K., "Convolution back-projection image reconstruction for Synthetic Aperture," *Proc. IEEE International Conference on Circuits and Systems*, Montreal, Canada, May 1984.
- [18] Desai M. and Jenkins W. K., "Performance evaluation and architectural implications of convolution back-projection image reconstruction for synthetic aperture," *Proc. IEEE International Conference on Circuits and Systems*, Tokyo, Japan, May 1985.
- [19] Elachi C., Bicknell T., Jordan R. L., Wu C., "Spaceborne synthetic-aperture imaging radars: applications, techniques and technology," *Proc. IEEE*, vol. 70, no. 10, October 1982, pp. 1174-1208.
- [20] Hanson, "Computed tomographic (CT) reconstruction from limited projection angles," *SPIE Application of Optical Instrumentation in Medicine*, vol. 347, 1982, pp. 166-171.
- [21] Harger R. O., *Synthetic aperture radar systems*. New York: Academic Press, 1970.
- [22] Hayner D. A., "The missing cone problem in computer tomography and a model for interpolation in synthetic aperture radar," Ph.D. thesis University of Illinois, 1983.
- [23] Hildebrand B. P., "Statistics of focused and defocused maps," *IEEE Trans. on Aerospace and Electronic Systems*, vol. AES-10, no. 5, September 1974, pp. 615-621.
- [24] Inouye, T., "Image reconstruction with limited angle projection data," *IEEE Trans. on Nuclear Science*, vol. NS-26, no. 2, April 1979, pp. 2666-2669.
- [25] Jenkins W. K., "Hardware architectures for complex digital signal processors based on quadratic number codes," *Proc. IEEE International Conference on Circuits and Systems*, vol. 2, pp. 755-758, Tokyo, Japan, May 1985.
- [26] Jenkins W. K., "Quadratic modular number codes for complex digital signal processing," *Proc. IEEE International Conference on Circuits and Systems*, vol. 1, pp. 264-267, Montreal, Canada, May 1984.

AD-A170 652

A NEW METHOD OF SYNTHETIC APERTURE RADAR IMAGE  
RECONSTRUCTION USING MODIF. (U) ILLINOIS UNIV AT URBANA  
COORDINATED SCIENCE LAB N D DESAI AUG 86  
UILU-ENG-86-2225 N00014-84-C-0149

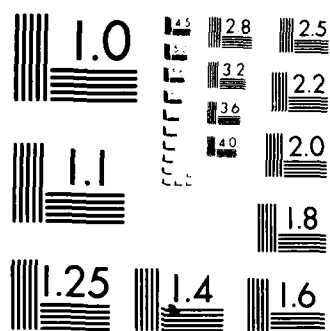
2/2

UNCLASSIFIED

F/G 17/9

NL





MICROCOPY RESOLUTION TEST CHART  
NATIONAL BUREAU OF STANDARDS-1963-A

- [27] Jenkins W. K. and Desai M., "The discrete-frequency Fourier transform: a missing tool in digital signal processing ." *Proc. IEEE International Conference on Circuits and Systems*, Tokyo, Japan, May 1985.
- [28] Jenkins W. K., Mather B., and Munson D. C. Jr., "Nearest Neighbor and generalized inverse distance interpolation for Fourier domain image reconstruction." *Proc. IEEE International Conference on Acoustic Speech and Signal Processing*, Tampa, Florida, March 1985.
- [29] Johnson R. B., "Radar Analogies for Optics," *Proc. Effective Utilization of Optics in Radar Systems*, SPIE vol. 128 1977, pp. 75-83.
- [30] Kak A. C., "Algorithms for reconstruction," *Proc. Cardiovascular Imaging and Image Processing Theory and Practice* pp. 163-175, July 1975.
- [31] Kirk J. C., Jr., "A discussion of digital processing in SAR," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 11, pp. 326-337, May 1975.
- [32] Kock W. E., *Radar Sonar and Holography*. New York: Academic Press, 1973.
- [33] Kogge P. M., *The Architecture of Pipelined Computers*. New York: McGraw-Hill Book Company, 1981.
- [34] Kovaly J. J., *Synthetic Aperture Radar*. New York: Academic Press, 1970.
- [35] Krogmeier J. V. and Jenkins W. K., "Error detection and correction in quadratic residue number systems," *Proc. 26 th Midwest Symposium on Circuits and Systems*, pp. 408-411, Puebla, Mexico, August 1983.
- [36] Kung H. T. "Why systolic architecture?", *Computer Magazine* vol. 15, no. 1, January 1982, pp. 37-46.
- [37] Larson R. W., Zelenka J., and Johansen E. L., "Microwave hologram radar system," *IEEE Trans. on Aerospace and Electronic Systems*, vol. AES-8, no. 2, March 1972, pp. 208-216.
- [38] Lee H. and Wade G., "High-resolution imaging for systems with small apertures," *Journal of Accoust. Soc. Am.* vol. 72, no. 6, December 1982, pp. 2033-2034.
- [39] Li F. K., Held D. N., Curlander J. C., and Wu C., "Doppler parameter estimation for spaceborn synthetic-aperture radars," *IEEE Trans. on Geoscience and Remote Sensing*, vol. GE-23, no. 1, January 1985, pp. 47-55.
- [40] Leith E., "Quasi-holographic technique in the microwave region," *Proc. of IEEE*, vol. 59, no. 9, September 1971, pp. 1305-1318.

- [41] Mehrlis J. G., "Synthetic aperture radar range-azimuth ambiguity design and constraints," *IEEE International Radar Conference*, pp. 143-146.
- [42] Miller V. C., *Photogeology*, New York : McGraw-Hill, 1961.
- [43] Mersereau, R. M., "Direct Fourier transform techniques in 3-D image reconstruction," *Comput. Bio. Med.*, Pergamon Press 1976, vol. 6, pp 247-258.
- [44] Mersereau R. M., Oppenheim A. V., "Digital reconstruction of multidimensional signals from their projections," *Proc. IEEE*, vol. 62, Oct 1974, pp. 1319-1338.
- [45] Munson D. C. Jr., O'Brien J. D., and Jenkins W. K., "A Tomographic formulation of spotlight mode synthetic aperture radar," *Proc. IEEE*, vol. 72, August 1983, pp. 917-925.
- [46] Munson D. C., Jr., Sanz J. L. C., "Image reconstruction from frequency-offset fourier data," *Proc. IEEE*, vol. 72, no. 6, pp. 661-669, June 1984.
- [47] D. C. Munson, Jr., Jorge L. C. Sanz, W. K. Jenkins, Gary Kakazu and Bruce Mather, "A comparison of algorithms for polar-to-cartesian interpolation in spotlight mode SAR," *Proc. IEEE International Conference on Acoustic Speech and Signal Processing*, Tampa, Florida, March 1985.
- [48] O'Brien J. D., "A consideration of signal processing for spotlight synthetic aperture radar," Ph.D. thesis, University of Illinois, 1983.
- [49] Platzer, Etschberger "Fouriertransformation zweidimensionaler signale," *Separatdruck aus Laser+Elektro-Optik* pp. 1-16, 1972.
- [50] Rihaczek A. W., *Principle of High Resolution Radar*, New York: McGraw-Hill, 1969.
- [51] Schafer R. W. and Rabiner L. R., "A digital signal processing approach to interpolation," *Proc. IEEE*, vol. 61, pp. 692-702, June 1973.
- [52] Schwartz D. A., "Analysis and experimental investigation of three synthetic aperture radar formats," Technical Report T-94, Coordinated Science Laboratory, Univ. of Illinois, Mar. 1980.
- [53] Scudder H. J., "Introduction to computer aided tomography," *Proc. IEEE*, vol. 66, pp. 628-637, June 1978.
- [54] Walker J. L., "Range-doppler imaging of rotating objects," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 16, pp. 23-52, Jan. 1980.

- [55] Weis W. G. and Jenkins W. K., "Theory and simulation of digital FFT processors for spotlight mode synthetic aperture radar." *Lockheed Aircraft Corporation Report.. LMSC/D566409*, December 1977.

END

DTIC

9-86